MOJ MIKRO

TELEKS

RADIO ŠTUDENT

december 1984 / cena 200 dinarjev

ČE ŽE RAČUNATE,
RAČUNAJTE S POMOČJO
SHARPA
PREDSTAVLJAMO VAM DVA MODELA KALKULATORJA S TRAKOM EL 1611, CENA
80 DM IN OKROG 65% DAJATEV, EL 2607, cena 170 DM IN OKROG 65% DAJATEV
KALKULATORJA S TRAKOM STA PRIMERNA ZA OSEBNO UPORABO.
SHARP
EL-1611
EL-2607
ZASTOPA IN PRODAJA M-MEDNARODNA TRGOVINA,
CONTAL
LJUBLJANA, TITOVA 65

MOJ MIKRO

Risba na naslovni strani: Zletko Drcar

December 1984

*E**ppur si muove . . .** Galileo Galilei je te besede menda izustil mrmraje, in tudi mi jih zapišimo zadržano – na področju računalniškega opismenjevanja se je resda nekaj premaknilo, vendar še ni razloga, da bi slavili.*

Veseli nas recimo novica, da bodo podravske šole, ki jih je približno 110, dobile 253 mikroračunalnikov in precej dodatne opreme (tiskalnikov, kasetofonov, diskovnih enot, igralnih palic itd.). Akcija je stekla zunaj okvirov znane republiške pobude »2000 računalnikov v slovenske šole«, s pomočjo družbenopolitičnih organizacij in krajevnega gospodarstva, ki bo zbralo več kot polovico denarja, potrebnega za nakup commodorjev (za ta model so se odločili, ko so tehtali razmerje cena – zmogljivost).

Novica je toliko važnejša, če vemo, da se v drugih republikah dijaki celo na srednješolski stopnji prerivajo pred enim samim zaslonom (v Beogradu, recimo, bodoče računalnikarje v usmerjenem izobraževanju ne učijo niti – angleščine). Nasploh je v drugih republikah opaziti, da je računalnik za zdaj le orodje navdušencev in statusni simbol premožnih ljudi (na veliko po vojvodinskih in pomoravskih vaseh).

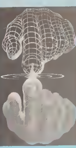
Nič čudnega, če zanimanje za računalnike tudi v marsikaterem uglednem družbenopolitičnem forumu (za Slovenijo to ne velja) gladko obravnavajo kot »modno muho«. Mar ni celo predsednica ZIS Milka Planinc primerjala hackerjev z rockerji? Ko je na nedavnem srečanju slovenskih novinarjev v Radencih odgovarjala na vprašanja, kdaj bo moč uvažati računalnike, je mladim gladko svetovala, naj se po zgledu ljubiteljev rocka povezujejo v klube in po tej poti zaobidejo prepoved uvoza.

Baterija bobnov seveda ni isto kot računalniški hardware. Poleg zastarelih zakonskih predpisov bo torej treba spremeniti način mišljenja. Pa tudi podravskim šolam dragocena strojna oprema ne bo kaj dosti pomagala, če z usklajenim delom na republiški ravni ne bomo zagotovili še dovolj ustrezne programske opreme, vključno s sposobnimi mentorji.

Mojemu mikru ni treba zardevati: ta mesec je postal prva jugoslovanska revija za vse ljubitelje in uporabnike mikroračunalnikov. Srbskohrvatski izdaji na pot pa smo v uvodniku zapisali: Mojega mikra ne pripravljamo samo v redakciji, naši bralci nam ne pošiljajo samo malih oglasov in pisem, temveč sodelujejo tudi z lastnimi programi in drugimi prispevki. Poziv ša vedno velja tudi za slovensko izdajo – samo skupaj bomo dosegli, da se na tem pomembnem področju ne bo le premaknilo, temveč da se bo premikalo. Vedno hitreje, neustavljivo.

Iz vsebine prihodnje številke

- Januarska številka bo izjemoma izšla že pred novoletnimi prazniki. Nanjo še posebej opozarjamo vse ljubitelje računalniških iger: zanje bomo pripravili pravcato prilogo, saj je čas praznikov tudi čas zabave.
- Informatika čedalje bolj prodira tudi v turizem. Z dvodnevnega posvetovanja v Radencih bomo izčrpno poročali o načrtih slovenskega gospodarstva.



- Tudi priloga bo v znamenju novega leta, s programom za stoletni koledar.
- Testi: HP 9000, računalnik vaših sanj
- Čudoviti svet dodatkov igralne palice

MOJ MIKRO pripravljata uredništvo revije TELEKS in software redakcija RADIO ŠTUDENT ● Izhaja vsak prvi torek v mesecu ● Izdaja in tiska ČGP DELO, tozd Revije, Titova 35, Ljubljana ● Predsednik skupščine ČGP Delo JAK KOPRIVC ● Glavni urednik ČGP Delo BORIS DOLNIČAR ● Direktor tozd Revije BERNARDA RAKOVEC ● Glavni in odgovorni urednik revije Teleks VILKO NOVAK ● Uredniki: CIRIL KRAŠEVEC, ŽIGA TURK in ALJOŠA VREČAR ● Oblikovanje in tehnično urejanje: ANDREJ MAVSAR, FRANCI MIHEVC ● Naslov uredništev: Teleks, Ljubljana, Titova 35, telefon 319-280, Radio Študent, Ljubljana, Študentsko naselje, blok 8, telefon 261-985 ● Oglasi: STIK, oglasno trženje, Ljubljana, Titova 35, telefon 318-570 ● Prodaja: Ljubljana, Titova 35, telefon h. c. 315-366 ● Cena številke 200 din ● MOJ MIKRO je oproščen plačila posebnega davka po mnenju republiškega komiteja za informiranje, dopis št. 421-1/72 z dne 25. 5. 1984

Commodore 64

Mavrični pogled na CBM-64

ŽIGA TURK

Lepe denarce sem odštel za svoj novi računalnik in priznati moram, ni mi žal. Že na prvih straneh navodil za uporabo so mi čestitali. Res sem se pametno odločil, saj sem kupil »enega najboljših mikroračunalnikov na svetu«. Trgovec mi je svetoval, naj si kupim še disketni pogon enoto VIC 1541. Tako bom imel doma zares profesionalen sistem, ki mi ga ne bo sram pokazati prijateljem. Dobro mi je svetoval, saj mi »User's Manual« želi dobrodošlico k »najhitrejšem, najenostavnejšem in najučinkovitejšemu načinu shranjevanja podatkov, ki je na voljo za Commodorove računalnike«.

Priznam, kriv sem. Že skoraj tri leta se mučim v glavnem z neuglednimi, gumijasto lepljivimi umotvori dragega strička Clivea, ki si vsake toliko privošči kakšno manjšo neumnost uporabniki njegovih mikroračunalnikov pa jo potem obesijo na veliki zvon. Šalo na strani Računalnika nisem kupil, ampak ga je uredništvu Mojega mikra posodil Konim iz Ljubljane, dinamična delovna organizacija, ki jugoslovanskemu kupcu omogoča, da za (relativno) majhen denar kupi (relativno) dobra računalnike.

Naj mi komodorjevci oprostijo, če bom v tej predstavitvi pozabil na kakšno pohvalo. Tega, da sem pozabil kaj pograjati, gotovo ne bodo priznali, saj so jih že navodila k računalniku prepričala, da denarja niso vrgli stran.

Commodore je tipična uspešnica mikroračunalniške revolucije. Iz newyorške delavnice, kjer so v petdesetih letih popravljali pisalne stroje, se je razvil četrti najmočnejši proizvajalec mikroračunalnikov na svetu (Apple, IBM in NEC so prva trojica). Pot navzgor se je začela s popularnim Commodorovim modelom PET, domačo živalco iz časa (in sence) prvih jabolk. Leta 1982 je bil v Ameriki računalnik leta VIC 20, prvi barvni mikroračunalnik za sprejemljivo ceno. V primerjavi z ZX spectrumom, ki mu je v Angliji sledil le nekaj mesecev pozneje, pa je bil s 24 znaki v vrstici in s črkami za slabovidne naravnost smešen. Ime VIC je dobil po posebnem čipu (Video Interface Chip), ki je procesor rešil napornih nalog ob generiranju slika.

Podoben čip uporablja tudi C-64 (VIC-II-6567). Commodore je namreč član istega koncerna kot proizvajalec mikroprocesorjev MOS Technology (6502, 6510). V ZDA ima tovarna nekako tak sloves kot Sinclair v Evropi. Tržišču so ponudili računalniški volkswagen, tudi za ceno kvalitete. Menda se vsak tretji C-64 vrne v trgovino (revija Creative Computing, febr. 1984), servisna dejavnost pa je pregovorno slaba.

Blišč

Vtis, da imam vendarle v rokah »zares« računalnik, je še podkrepila lepo oblikovana tipkovnica s profesionalnimi tipkami, na katere je veselje pritiskati. Razporeditev tipk je mestoma nekoliko čudna, zato pa so znaki za vse osnovne matematične operacije dostopni brez pritiskanja na tipko SHIFT. Pogrešal pa sem enakopravnost kazalčnih tipk. Ista tipka za premikanje je namreč za gor in dol, druga pa za levo in desno. Med smerama izberemo s tipko SHIFT, vendar se do danes še nisem navadil, za katero stran je treba »šiftati«, in zato vedno sprotI poskusim. Tudi avtomatsko ponavljanje tipka bi se prileglo, a to je samo eden od manjših softvarskih problemov.

Tokrat ne bo prav nobene potrebe po kakšni povezavi med tem, kaj znajo programi in kaj zmore strojna oprema. Ameriški propagandisti bi za to našli izraz »machine independant programming«. Računalnik skriva v notranjosti 36 čipov, od tega pomeni 8 pomnilniških čipov, ki zagotavljajo bleščačo štirišestdesetico v imenu, manjši del stroškov. Čipov je torej skoraj enkrat več kot v spectrumu. To je sicer zabavno za samograditelje in serviserje, poveča pa stroške izdelave. Število čipov bi bilo še večje, če ne bi vdelali posebnih čipov za kontrolo periferních enot. Omenili smo že VIC, ki skrbi za sliko. Ta ima 320 točk po širini in 200 po višini. Razlika od zaslona ZX spectruma je kar občutna. Barve so določene podobno kot pri spectrumu, z atributi. C-64 ima 16 različnih barv, vendar brez utripanja in poudarjene svetlosti. Strojna oprema podpira tudi 8 sličic (angl. sprite) velikosti 21*24 točk, ki pa jih lahko povečujemo. To pomeni, da nam bo strojna oprema sama skrbela za premikanje napadalcev iz vesolja. Kakšne druge uporabnosti pa v njih zaradi konstantne velikosti na vidim.

Poseben čip kontrolira tudi tipkovnico in vrata za igralno palico. Čip SID (Sound Interface Device) skrbi za zares bogato paleto zvokov: dva generatorja tonov in generator šumov. Izbiramo lahko glasnost zvoka, obliko zvočnih valov in štiri druge parametre, s katerimi določamo lastnosti glasbenega instrumenta, ki ga bomo posnemali. Učinki, ki jih je mogoče iz tega čipa izvabiti, so neverjetni, in če bi imel računalnik dalj časa, bi mi verjetno postalo žal, zakaj na vem več o glasbi.

Procesor je 6510 (=6502+8 bit v/i vrata), čistokrvni 8-bitni revček, popolnoma brez spogledovanja s 16-bitnimi instrukcijami, je pa zato v strojnem programiranju vsaj ostalo nekaj več tiste hekerske napetosti.

Na zadnji strani računalnika so priključki za dodatke: kartico ROM, antenski izhod za TV, avdio/video vtičnica (na video sliki se nisem ravno urezal), serijski (!) vmesnik za disketno enoto ali tiskalnik, vmesnik za kasetofon (kupiti boste morali Commodorovega in ne kar tistega, na katerem poslušate Avsenike) in razširitvena vrata, na katerih lahko spreten samograditelj odkrije marsikaj koristnega.

Na desni strani so še dve vtičnici za igralne palice in stikalo za vklop/izklop računalnika. Tipka RESET tudi CMB-64 nima, vendar smo v MM novembra objavili načrt, kako jo vdelati. V tolažbo vsem, ki se pritožujete nad nestandardnostjo svojega računalnika, naj povem, da je tu VSE nestandardno (razen konektorjev za igralne palice). To kaže na dosledno spoštovanje filozofije: »Tistega, kar mi lahko dodaš, na kupuj drugje!«

Potemkinov pomnilnik

C-64 naj bi bil prvi računalnik s 64 K pomnilnika za v standardni izvedbi. Danes, ko razmišljamo, ali je 256 K RAM v IBM-PC dovolj, ali ne, številka več na zbuje občudujočega vzdihovanja. Pogled na karto pomnilnika pa nas postavi na trdna tla. 1 K zavzamejo sistemske spremenljivke (poznavanja teh odloča, ali boste sami

Nadaljevanje na strani 12

Integrirano vezje Commodorja 64:

1. 26528 vmesna čipa kontrolirata povezavo med 6510 procesorjem in tipkovnico, igralnimi palicami, uro in dodatki.
2. 8K basic ROM
3. 8K operacijski sistem
4. 4K ROM za nabor znakov.
5. 6510 centralna procesna enota, možgani Commodorja 64.
6. Pretvarnik video signalov za TV izhod
7. Vrata za programsko opremo v ROM
8. 8 spominskih čipov po 8K zlogov vsak
9. 6581 čip za generiranje zvoka (SID)
10. Aluminijasta plošča ščiti čip, ki generira sliko (VIC)

MOJ MIKRO
TEST
SHARP
PC-1500A

Sharp PC-1500A

CIRIL KRAŠEVEC

Vse, kar imam, nosim v žepu: Dva roli papirja, rezervne baterija, robec in računalnik PC-1500 A. Priznam, zdaj ko je zunaj mraz, največkrat uporabljam robec. Večkrat pa sem tudi potežkal računalnik in previdno pritisnil kakšno tipko.

Računalnik je namenjen poslovnežem, ki želijo imeti povsod s seboj stroj, ki jim ga je brzkone kdo sprogramiral, zdaj pa ga koristno uporabljajo. Morda je namenjen tudi študentom ali celo naprednim učencem, ki si ga tako zelo želijo, da bi si vanj shranjevali »plonkcegice« (informacija iz avtobusa)? Ali pa predvsem ljudem, ki so jim všeč majhne japonske igrače. Gotovo je bil namenjen tudi nam, saj nam je pojedel nekaj popoldnevov. Morda jih bo še komu?

Odlična zasnova

PC 1500 A je v bistvu žepni računalnik izrednih zmogljivosti, ki nam lahko rabi kot podaljšek osebnega računalnika. Z vmesnikom CE-158 se lahko priključuje na domači računalnik, modem ali večji tiskalnik. Brez tega dodatka pa ga lahko uporabljamo kot žepni računalnik ali kot namizni računalnik s tiskalnikom in kasetofonom za zunanji pomnilnik. Napajamo ga lahko z omrežno napetostjo prek adapterja ali pa s štirimi baterijami 1,5 V.

Izredno lično ohišje zakriva goro elektronike, japonske duhovitosti in domiselnosti. Tipkovnica lepa in natančna, predvsem pa

namenjena ljudem tankih prstov, dala igrači primerno. Zaslon LCD s 26 znaki nam omogoča izpis vrstice in rezultatov programa. Štiribarvni risalnik pa nam rezultate napiše in izriše na ravno tako miniaturen papir, kot je zaslon. Na displayu lahko pišemo mala in velike črke, številke, matematične in posebne znake ter znake, ki jih definiramo sami. Na zaslonu 7×156 točk smo streljali celo invaderje.

Basic

Malo čudo lahko programiramo v pravem basicu brez kakšnih posebnosti. Operiramo lahko s številskimi vrednostmi, dopuščenimi z 10 mesti za mantiso in dvema mestoma za eksponent. Spremenljivke so lahko dolga dva črki. Polja so lahko dvodimenzionalna.

Med ukazi najdemo celo ON ERORR GO TO in TRACE. Pri prvem se program ob napaki

nadaljuje v navedeni vrstici. Pri drugem pa napisani program v basicu izvajamo kakor za korakom in sledimo izvajanju na displayu.

V naboru ukazov je nekaj takih, ki jih pod temi imeni ne srečujemo prav pogosto. Prvi je ukaz USING, ki določa, koliko mestnimi števili oz. na koliko decimalk natančno bomo izpisovali rezultate. Potem sta tu CURSOR in GCURSOR, prvi za izpis na zaslonu, drugi na tiskalniku. V bistvu je to nadomestek za običajno funkcijo TAB. Omogočeno je tudi delo z grafičnimi znaki (nizi). V ta namen so na razpolago ukazi LEFT$ in MID$.

Posebej se kaže pomuditi pri ukazih za delo z risalnikom. Najprej je tu TEST, ki nam nariše štiri raznobarvne kvadratke 5×5 mm, nato pa še ukazi: LF za pomik papirja navzgor, CSIZE za velikost črk, COLOR za barvo peresa in ROTATE za smer pisanja.

Prav ukaz ROTATE nas je presenetil, saj z njim enostavno pišemo v katerikoli smeri. Tako je bila z zelo kratkim programom narejena tudi glava za naš test.

Ukazi za delo s kasetofonom so običajni, le da jim je dodana prva črka C. Tako dobimo ukaza CLOAD in CSAVE. Ker ima malček tudi daljinsko kontrolo nad dvema kasetofonoma, lahko z dopolnjenim ukazom CLOAD-1 ali CLOAD-0 beremo z drugega ali s prvega priključenega kasetofona. To velja seveda samo za originalne ali združljive kasetofone.

Vse ukaze tipkamo črko po črko, čeprav je nekaj tudi že definiranih ukazov (keywords). Do teh ukazov pridamo s tipko DEF in pritiska na določeno tipko. Definirani ukazi so: INPUT, PRINT, USING, GOTO, GOSUB, RETURN, CSAVE, CLOAD, MERGE in LIST.

Dodatek za uporabo kasetofona in risalnik

CE-150 je vmesnik, ki ima na eni strani miniaturno izvedbo tiskalnika risalnika, ki smo ga spoznali že pri Sharpovem modelu MZ-700, na drugi pa priključke za standardni kasetofon. Širina papirja za risalnik je standardna, 58 mm, to pomeni, da lahko uporabljamo tudi boljši papir za registrske blagajne, ki ga dobimo pri nas.

Na čelni plošči so samo štirje gumbi. Z enim vklapljamo in izklapljamo daljinsko kontrolo kasetofona. Drugi trije so namenjeni risalniku. Eden je za resetiranje risalnika in pomik papirja naprej. Drugi nam samo odklene pokrov risalnika, če menjamo papir. S tretjim lahko onemogočimo izpis na papirju med izvajanjem programa.

Na zadnji strani vmesnika je še konektor za razširitve. To je konektor z istimi priključki kot na samem računalniku, ki pa smo ga s priključitvijo na vmesnik CE-150 zasedli.

Računalnik in vmesnik še vedno lahko napajamo iz baterij računalnika ali prek napajalnika iz omrežja. Priključek je zdaj na desni strani, poleg vtičnic za kasetofone.

Rami, ki ne pozabljajo

Še ena japonska rami, ki ne pozabljajo. Kakšno svetovno čudo je to? Pomnilnik RAM vendar, ko izključimo računalnik, vse pozabi. PC-1500A lahko izključimo in ponovno prižgemo, pa nam vse ostane v pomnilniku tako, kot smo pustili. Seveda ko ima baterija. In kaj pravite na to, da priključimo na konektor na spodnji strani 16 K RAM, napišemo program, iztaknemo RAM, ga spravimo v žep, čez kak dan pa spet priključimo in popravljamo isti program?

To so tako imenovani programski moduli z baterijo. V plastičnem ohišju je poleg rama baterija iz litija, ki približno pet let skrbi za vseh 16 384 bytkov. Takšni moduli so dosegljivi v dveh inačicah, 16 in 8 K. Lahko pa si omislimo module RAM brez baterije (8 in 4 K). Na ravno takšnih modulih dobite tudi trajno vpisane programe za statistiko in razna praktična izračunavanja.

Povezava z zunanjim svetom

V te namene so pri Sharpu izdelali že omenjeni vmesnik CE-158. Ta posreduje v računalnik in iz

Nadaljevanje na str. 11

```
10:FOR I=0TO 29
20:READ A
30:POKE &5900+I,A
40:NEXT I
50:DATA 72,56,74,
   197,68,5,183,1
   3,153,6,68,5,1
   83,255,153,12,
   132,174
60:DATA 120,103,4
   ,174,120,104,1
   81,0,174,56,19
   7,154
70:END
```

```
PROGRAM        &5900
               &591D
CODE           &5900
               &591D
TABELA SIMBOLOV:
NASLOV         &5904
PROGRAMSKE KODE:
5900: 48 52 4E B8 44 85 87 00
5908  95 68 44 85 B7 FF 99 0C
5910  84 AE 78 67 84 AE 78 68
5918  B5 AA AE 52 68 9A
M=5900
5900 LD XH 38            48
5902 LD XL,C5            4A
5904 INC X               44
5905 LD A,(X)            85
5906 CP A,00             B7
5908 JR NZ,5904          99
590A INC X               44
590C CP A,FF             B7
590E JR NZ,5904          99
5910 LD A,XH             84
5911 LD (7867),A         AE
5914 LD A,XL             84
5915 LD (7868),A         AE
5918 LD A,00             B5
591B LD (78C5),A         AE
591D RET                 9A
```

Kako spraviti v tek program, zbrisan z ukazom NEW? Odgovor je program pred vami, pisan v strojnem jeziku.

Z ukazom NEW postavimo na začetne vrednosti le BASIC END-POINTER. To pa pomeni, da je program še vedno v spominu, le v drugačni obliki. Po izvedenem ukazu NEW dobite program v bazicu zopet v spomin, če vpišete CALL &5900. Seveda moramo vtipkani program najprej startati z RUN, da se vpišejo vrednosti strojnega programa.

Strojni program lahko shranimo naposredno na kaseto s CSAVE M »ime« &5900 &591D. V računalnik pa ga vpišemo s CLOAD M »ime« &5900.

Andrej Prazalj, Prušnikova 13, Šentvid

Od abaka do čipa

IVAN BERGLEZ

Potreba po pripomočkih za računanja je stara kot človeška civilizacija. S spološnim napredkom tehnike so se izpopolnjevala tudi naprave za računanja.

V 17. stoletju so bili že resni poskusi, da bi izdelali mehanske računske stroje (Pascal, Leibniz).

V začetku 19. stoletja so stroji še vedno mehanični, ideje za gradnjo pa že presegajo tehnične možnosti (Babbage, 1833).

V drugi polovici 19. stoletja omogoča elektrika realizacijo principov, ki jih prej mehanika ni mogla.

Obdelava podatkov se začne leta 1890 v ZDA, ko obdelajo podatke ljudskega štetja. Od takrat je razvoj izredno hiter.

Zanimivo je, da so najprej razvili večje računalnike, in šele po letu 1972 se pojavijo mikroračunalniki za različna namena.

Na razvoj računalnikov je znatno vplival razvoj drugih panog, predvsem matematike in fizike. V nadaljevanju so po časovnem zaporedju podani pomembnejši mejniki in dogodki, ki so prispevali k razvoju te panoge od prvih začetkov do današnjih dni.

5000 pred našim štetjem

Prvi začetki številskih znakov so bili prsti na rokah in nogah: od tod pozneje simbolna znamenja.

3000 pr. n. š.

Pri Sumercih nastajajo številski znaki in številski sistem z osnovo 60, ki ga pri merjenju časa poznamo še danes (ura = 60 minut, minuta = 60 sekund).

2000 pr. n. š.

V Egiptu se pojavi slikovna predstavitev znakov, pri Babiloncih pa klinopis. Nobena od teh pri številih ne pozna mestnih vrednosti (enice, desetice itd.). Zato za večja števila uporabljajo posebne simbole.

1100 pr. n. š.

V začetku dinastije Ču (Kitajska) nastane preprosta naprava za računanje s kroglicami na palčkah suan-pan. Podobne naprave nastanejo pozneje tudi pri Japoncih (soroban), Rusih (sčoti), Rimljanih (abak). Te naprave že poznajo mestno vrednost.

330 pr. n. š.

Grški filozof Aristotel razvija logiko sklepanja in dokazovanja.

62

Heron iz Aleksandrije naredi prvi programiran avtomat, ki odpira vrata templja.

800

Arabci spremenijo indijske znake v obliko, ki jo poznamo še danes. Uporabljajo tudi ničlo, ki je Rimljani niso poznali.

820

Arabski matematik Alhvarazvi napiše knjigo o indijskih številkah, kar znatno razširi pojmovanje o številkah in številih. Iz njegovega imena je izpeljan pojem algoritem.

16. stoletje

Arabske številke se pojavijo konec tega stoletja.

Švicar Jost Burgin izdela kroglo, na kateri je možno samodejno določiti položaj čez 1000 zvezd. Pripisujemo mu tudi iznajdbo logaritmov (ki jih nato že leta 1605 uporablja Kepler). Ker pa jih objavi šele leta 1620, je odkritje priznano škotskemu matematiku Johnu Napieru, ki jih je objavil leta 1614 in 1619.

1620

Napier izdela prvo logaritmično računalo.

1623/24

Wilhelm Schickerd zgradi prvi računski stroj, vendar se ta izgubi v 30-letni vojni. Pripisujejo mu tudi izum mehanskega števca s prenosom desetic.

1642

Blaise Pascal zgradi stroj za seštevanje 8-mestnih števil. Posredno je bilo možno tudi odštevanje. Leta 1647 zgradi podoben stroj za 8-mestna števila. Oba stroja delata na principu prenosa zobatih koles.

1673

Wilhelm Leibniz izdela računski stroj (Replica), ki zmore 4 osnovne računske operacije z 12 mesti. S tem je izdelan osnovni princip mehaničnega računskega stroja. Model tega stroja hrani Nemški muzej v Münchnu. Leibniz odkrije tudi dvojiški številski sistem, ki postane pozneje osnova elektronskih računalnikov.

1728

Francoski mehanik Falcon uporabi za krmiljenje tkalskega stroja preluknjane deščice.

1805

Joseph Marie Jacquard izpopolni idejo krmiljenja na svojem tkalskem stroju in uporabi papirnato kartico. To omogoči izdelavo velike količine blaga z enakomerno kakovostjo. Princip krmiljenja s preluknjanim trakom postane uporaben na različnih strojih tistega časa.

1822

Charles Babbage (angleški matematik 1792–1871), izdela diferenčni stroj za izračun logaritmov. Z njim lahko računa razliko dveh vrednosti in razliko razlik na 8 decimalnih mest.

Zasnuje tudi načrt diferenčnega stroja, ki bi lahko računal razliko 5. stopnje na 20 decimalnih mestih. Vendar izdelava mehanskega pogona ni bila tehnično izvedljiva.

Logaritmično računalo v močnejši izvedbi

1833

Charles Babbage izdela načrt analitskega stroja, ki ima v principu vse enote 100 let poznaje zgrajenih računalnikov: aritmetično enoto, pomnilnik, krmilno enoto in vhodnoizhodne enote.

Tehnična sredstva tistega časa niso omogočala izdelavo takega stroja (elektrike še niso poznali). Po načinu razmišljanja je Babbage daleč prehitel sodobnike, ki ga niso razumeli. V veliko pomoč mu je bila grofica Ada Augusta (1815–1852), ki je objavljala njegova dela in tako širila njegove zamisli. Njej v čast se imenuje v zadnjem času razviti programski jezik ADA.

1847

George Boole (1815–1864) razvije posebno vrsto algebre, ki omogoča matematično obdelavo logičnih trditev. Ta zvrst dobi po njem naziv Boolova algebra. Koristna je pri izdelavi optimalnega vezja za določeno funkcijo.

1873

Tovarna Remington izdela prvi pisalni stroj za komercialne namene (Remington No. 1).

1874

Ferdinand Braun (profesor v Leipzigu) odkrije naravno polprevodnost na kristalih. Nekateri kristali namreč prepuščajo tok v eno smer, v drugo pa ne. Kar odkritje ne najde praktične uporabe, gre v pozabo.

1890

Hermann Hollerith (1860–1929) izdela elektromehanično napravo (tabelirni stroj) za sortiranje. Na njej so obdelali podatke 11. ljudskega štetja v ZDA. Podatki so bili zluknjani v papirnati kartici velikosti 3×5,5 cole. Pri prejšnjem štetju je 50 delavcev obdelovalo podatke 7 let, Hollerith je z istim številom delavcev in s 43 stroji opravil delo v dobrem mesecu.

To leto imamo za začetek obdelave podatkov. Nosilec podatkov je luknjana kartica v posebni kodi, ki jo je izdelal Hollerith in se po njem tudi imenuje.

Leta 1896 ustanovi Hollerith družbo Tabulating Machine Company, ki se 1924 preimenuje v International Business Machines (IBM) in je danes na prvem mestu po proizvodnji velikih in malih sistemov.

1906

De Forest in R. von Lieben vdelata v katodno cev tretjo elektrodo, s katero dosežeta ojačenje izmeničnega toka. Leta 1910 je izdelana prva ojačevalka (trioda).

Hollerithova naprava za luknjanje in tabelirni stroj

1914

Siemens in Halske izdelata prvo vakuumsko elektronko za nizkonapetostno ojačenje.

1930

Davidor Schottky in Mott pojasnijo mehanizem polprevodnosti.

Tauschack izdela stroj za luknjanje podatkov v kartice, na katerem je možno izbirati funkcije s stikali na stikalni plošči.

1937

George Stibitz zgradi za družbo Bell Telephone Labs več računalnikov na osnovi relejev.

1938

C. E. Shannon in Shestakovs dokažeta, da je vse računske operacije možno izvesti s tremi osnovnimi logičnimi funkcijami IN, ALI, NE. To velja tudi za izjave da-ne v dvojiškem sistemu.

1940

V ZDA začnejo izdelovati monokristale silicija in germanija kot osnovo za polprevodnike, da bi zamenjali težke elektronke, ki porabljajo ogromno energije. Raziskave v tej smeri pripeljejo 8 let pozneje do odkritja tranzistorja.

1941

Nemec Konrad Zuse, ki je že leta 1938 zgradil mehanični računalnik z dvojiškim sistemom, zgradi za letalsko družbo računalnik Zuse 3 z naslednjimi značilnostmi:

– 600 telefonskih relejev v aritmetični enoti
– 3000 relejev v pomnilniku
– 15 do 20 operacij v sekundi s 7-mestnimi števili
– možen je izračun kvadratnega korena
– izhodna enota je plošča s signalnimi lučmi

Model tega računalnika hrani Nemški muzej v Munchnu.

1944

Howard H. Aiken (profesor matematike na univerzi Harvard v Cambridgeu) dokonča računalnik MARK 1, ki ga je začel graditi že leta 1939. Računalnik nosi tudi oznako ASCC, Automatic Sequence Controlled Calculator. Nekatere njegove značilnosti so:

– delo v desetiškem sistemu
– 760 000 posameznih delov
– množenje v 6 sekundah
– porabljeneh 8000 km žice
– deloval je do 1959 v ameriški mornarici

Zanimivo je, da sodobnika Zuse in Howard zaradi vojne nista vedela drug za drugega.

1945

Vsi do sedaj omenjani računalniki so dobivali ukaze od zunaj z luknjanih kartic ali trakov. Vejitve, znake in podprogrami so zahtevali dodatne vhodne enote. Tako so nekateri računalniki potrebovali nekaj 10 čitalcev za trak, ki jih je bilo treba vključevati in izključevati glede na program.

John von Neumann objavi princip o kodiranju krmilnega programa in vnos tega v sam pomnilnik računalnika.

1946

J. P. Eckart in J. W. Mauchly zamenjata releje s hitreje delujočimi elektronkami in zgradita računalnik ENIAC (Electronc Numerical Integrater And Computer). Delo je trajalo 3 leta na univerzi v Pensilvaniji, zgrajen pa je bil za Balistic Research Laboratory. Za primerjavo s poznejšimi računalniki pogosto navajamo njegovo obsežnost in zgradbo:

– 18 000 elektronk
– 70 000 uporov
– 10 000 kondenzatorjev
– 1500 relejev
– teža 30 ton
– površina 10×15 m
– poraba energije čez 150 KW
– seštevanje 0,2 tisočinke sekunde
– množenje 2,8 tisočinke sekunde
– delovanje v desetiškem sistemu

Delal je do leta 1955.

1948

Družba Bell Telephone Corporation predstavi v New Yorku novo odkritje – tranzistor (izraz je sestavljen iz besed transconductance in transresistance). Zaradi dobre kvalitete, nizke cene in majhne velikosti povzroči nagel razvoj na vseh področjih elektronike. Izdelali so ga: John Bardeen, William Bradford Shockley in Walter H. Brattin.

IBM predstavi prvi računalnik SSEC z naslednjimi značilnostmi:

– 12 500 elektronk
– 21 400 relejev
– dela v kodi BCD (binary coded decimal)

— 66 čitalcev trakov za krmiljanje
Hkrati razvije IBM računalnik IBM 604, ki je zgrajen samo iz elektronk

1949

John von Neumann in Hermann Goldstein z univerze v Princetonu izdelata računalnik, ki dobi več imen

Neumannov stroj
princetonski stroj
IAS (Institute for Advanced Study at Princeton)

Pozneje izdelajo še celo vrsto podobnih računalnikov z imeni Maniac, Silliac, Illiac, Audiac, Oracle in Johniac

Pod vodstvom M. V. Wilkesa izdelajo na univerzi v Cambridgeu računalnik EDSAC (Electric Delay Storage Automatic Computer) s programom v notranjem pomnilniku. Vdelana so tudi posamezna funkcije (množenja, korenjenja itd.)

Konstruktorji ENIAC izdelajo računalnik BINAC s popolno notranjo kontrolo

1950

Družbi Remington Rand Inc. se pridruži družba Eckert Mauchly Computer Corporation, ustanovljena po izdelavi računalnika ENIAC. Eckert in Maushly zgradita UNIVAC I, ki postane prvi komercialni računalnik. Prvi sistem je za naslednje leto instaliran v ameriškem statističnem uradu

Pri računalniku MARK III prvič uporabijo za zunanje nosilce podatkov magnetni boben in magnetni trak

o V istem času razvije Zuse sistem Z 4, ki ima dolžino besede 32 bitov, seštevanje opravi v 0,5 sek., množenje pa v 3,5 sek.

Na Institutu za tehnologijo v ameriški državi Massachusetts razvijajo računalnik WHIRLWIND I, s katerim se začenja serija miniračunalnikov. Njegovi ustvarjalci Kem Olsen, Dick Best in George Geralds pozneje ustanovijo družbo Digital Equipment Corporation (DEC)

1952

IBM izdela zelo hitri računalnik IBM 701, ki uporablja magnetne trakove

Na Institutu za tehnologijo v Massachusettsu izdelajo feritna jedra za pomnilnik. Že leta 1953 jih vdelajo v računalnik WHIRLWIND I

V tem letu postanejo tudi tranzistorji komercialno dosegljivi

1954

Backus in Ziller razvijeta programski jezik FORTRAN. IBM predstavi srednje veliki računalnik IBM 650

1955

Profesor Stiefel in sodelavca Speiser in Rutismauser izdelajo prvi švicarski računalnik ERMETH.

Pod vodstvom J. H. Felkerje izdela Bell Telephon Laboratory računalnik TRADIC, ki ima vdelane tudi tranzistorje

1956

IBM izdela računalnik IBM 305 RAMAC (random access memory), ki prvič uporabi magnetni disk (magnetno ploščo). Lincoln Laboratory izdela računalnik TX-O, v katerem so elektronka popolnoma zamenjana s tranzistorji (3600)

1957

Pri izdelavi računalnikov uporabijo koncept kanalov. Medtem ko daje računalnik rezultate prek enega kanala, lahko prek drugega včitava podatke za drugo nalogo. Po izstrelitvi sovjetskega satelita Sputnik 1 (4. 10. 1957) in ameriškega Explorerja 1 (31. 1. 1958) se prične zelo hiter razvoj mikroelektronike

Firmi Fairchaild se posreči izdelava tranzistorja na silicijevi osnovi z difuzijskim postopkom

V tem obdobju začenjajo izdelovati računalnike že mnoge evropske države (Anglija, Nemčija, Francija, Nizozemska, Italija, Švedska, Danska) in Japonska.

1958

Angleški fizik Brian Josephson poda osnove tehnologije za izdelavo integriranih vezij na ravni silicijevi ploščici (silicium planar technology)

Jack St. Clair Kilby (Texas Instruments) izdela različne komponente: tranzistor, upor, kondenzator, iz iste snovi. S tem omogoči nov način izdelave integriranega vezja

Tako vezje pride na trg že naslednje leto. Prva vezja so izdelane po bipolarni tehnologiji. Šele leta 1962 izdela Steven Holstein vezja na osnovi polprevodnih kovinskih oksidov (MOS)

1959

Po naročilu obrambnega ministrstva ZDA razvijajo pod vodstvom mornariškega kapitana Gracea Cooparja programski jezik COBOL

IBM izdela računalnik IBM 1401 in IBM 1620, ki sta popolnoma opremljena s tranzistorji

1960/61

Texas Instruments in Fairchaild imata za v redni proizvodnji integrirana vezja za komercialne namene

The IBM Market

Elmer Sperry

1961

Ob koncu tega leta je v uporabi čez 7000 računalnikov (leta 1957 samo 1300). Glavni proizvajalci so IBM, Sperry Univac, Burroughs, Control Data in Honeywell

1962

Sperry Univac izdela prvi komercialni računalnik s tankoplastnim pomnilnikom UNIVAC 1107

1963

Po svetu je instaliranih čez 16 000 računalnikov. Izdelani so tiskalniki, ki omogočajo izpis 600 vrstic v minuti

1964

IBM izdela sistem 360 z mikrostikalno tehniko, kar zmanjša razsežnost računalnika. Sistem se močno razširi po svetu. Tudi v Jugoslaviji se prične uvajanje računalništva prav s tem sistemom

Za potrebe vesoljskih poletov izdelajo miniaturne računalnike, npr. CDC 499. Njegova velikost je 10×10×23 cm

1965

Firma DEC pošlje na tržišče prvi komercialni procesni računalnik PDP, ki postane pozneje zelo razširjen tip računalnika

IBM serijsko izdeluje računalnik IBM 1800

1966

Sperry Univac izdela računalnik UNIVAC 9000

Za vnos podatkov se prvič uporabi optični čitalec

Na silicijevi ploščici je mozno izdelati vezje, ki obsega že 60 tranzistorjev

1966

Nekaj inženirjev tovarne Fairchild ustanovi INTEL

V tem obdobju je že možna izdelave integriranih vezij s 100 tranzistorji

1970

IBM izdela računalnik IBM 370 Prvič je omogočen neposredni pristop v pomnilnik

Sperry Univac izdela računalnik UNIVAC 9700

1970/71

Firma INTEL, ki je specializirana za izdelavo kompliciranih integriranih vezij, izdela prvi štiribitni mikroprocesor 4004 z 2300 tranzistorji

Že naslednje leto izdela osembitni mikroprocesor z 2900 tranzistorji. Rodi se mikroelektronika

1972

Na trgu se pojavijo žepni kalkulatorji in izpodrinejo logaritemska računala

1973

Sperry Univac izdela sisteme 90/60 in 90/70

1974/75

INTEL izdela znani mikroprocesor 8080 s 5000 tranzistorji

Izdelan je pomnilnik 4 K, ki vsebuje 2900 tranzistorjev ter množico kondenzatorjev in uporov v čipu

Hewlett-Packard da na tržišče prvi žepni računalnik HP 65, ki ga je možno programirati

Motorola izdela 8-bitni mikroprocesor M 6800, ki dela pri napetosti 5 V

Ob koncu 1974 predstavi National Semiconductors 16-bitni mikroprocesor. Ob koncu 1975 je na tržišču že okrog 40 različnih mikroprocesorjev

Skupina inženirjev firme INTEL ustanovi novo firmo ZILOG. Ta izdela znani mikroprocesor Z 80, izboljšano verzijo procesorja INTEL 8080

Hewlett Packard predstavi celo družino 16-bitnih mikroprocesorjev serije 9900

Ameriška firma MITS (Micro Instrumentation ad Telemetry System) prične prodajati komplete posameznih delov (kit), ki jih je mogoče sestaviti v funkcionalne enote

1976

INTEL izdela mikroračunalnik INTEL 8048 v enem čipu s pomnilnikom, ki omogoča vnos podatkov in izdajo rezultata

Odpirajo se trgovine z računalniki. Prva takih je The Computer Store v Los Angelesu (leta 1978 jih je že okrog 700)

Steve Jobs (24 let in Steve Wozniak (20) ustanovita tovarno Apple. Ta se v dveh letih uvrsti med vodilne proizvajalce mikroračunalnikov

1977

Pojavijo se pomnilniki z zmogljivostjo 16 K na enem čipu (20 000 tranzistorjev)

1978

Intel napove 16-bitni mikroprocesor 8086 (29 000 tranzistorjev). Sledijo mu Zilog z Z 8000 in Motorola z M 68 000, ki ima 32-bitno notranjo strukturo in je danes med najuspešnejšimi mikroprocesorji

IBM napove lastni 64 K čip, veliki računalnik IBM 3033 in sistem IBM 38

1979

Pri izdelavi integriranih vezij se začne obdobje zelo visoke integracije (VLSI). Na ploščico 20×40 mm je že položenih 100 000 tranzistorjev

IBM napove serijo računalnikov 4300. Sperry Univac pa 1100/60, ki je zgrajen v arhitekturi z mikroprocesorji

Firma Visicalc izdela znano programsko opremo, s katero opremljajo svoje računalnike različni proizvajalci

Intel objavi 1 MB pomnilnik na osnovi magnetnih mehurčkov

1980

Proizvajajo vezje, ki vsebujejo že 200 000 tranzistorjev

V tem letu prodajo okrog 120 milijonov mikroprocesorjev (60 odstotkov 4-bitnih, 39 odst. 8-bitnih, 1 odst. 16-bitnih)

1981

Intel zgradi 32-bitni mikroprocesor, ki obsega na treh gradnikih 160 000 tranzistorjev

1982

Sperry Univac napove največji računalnik sperry 1100/90

1983/84

Med velikimi sistemi ni bistvenih novosti. Izredno naraste število proizvajalcev mikroračunalnikov. Na tržišču je okrog 200 tipov mikroračunalnikov

Opomba. Pogosto govorimo o generacijah računalnikov. Ta delitev temelji na vrsti gradnikov, kar je razvidno iz tabele

Obd	Gradniki logičnih vezij	Generac
1940 1946	releji elektronke	1
1958	tranzistorji	2
1965	mikroelementi z integriranim vezjem	3
1970	mikroelementi na Si ploščici	4

Nadaljevanje s str. 7

njega podatke po dveh poteh. Prva je paralelna Centronicsova, druga pa serijska RS 232 C. Obe sta standardni. Nanju lahko priključimo vse zunanje naprave: tiskalnik, risalnik, modem in celo disketni pogon. Ta zadnje je malo privlečena za lase. Vendar je v tej stroki že res, da ni nič nemogoče razen nemogočega. Če bi po nevodilih popolnoma enostavno povezali PC-1500A s katerimkoli drugim računalnikom, bi ugotovili, da nismo napravili še ničesar.

Delo se v tej točki šele začne in največkrat so mu kos samo zelo dobri programerji. Edina res preprosta možnost je povezava z večjim Sharpovim računalnikom MZ-80B, kajti programe obeh računalnikov za takšno komuniciranje sta že napisane v priročniku

Programska oprema

Programske opreme za ta računalnik po svetu kar mrgoli. Najprej je treba pomisliti na močno propagando Sharpa v Zahodni Evropi, pa tudi na to, da na primer nemški starši v večini primerov precej laže ustrežejo svojemu mulcu z računalnikom za plonkanje. Programi so objavljeni skoraj v vsakem računalniškem časopisu in tudi oglasov za posebne uporabne programe z vseh področij znanosti in tehnike je dovolj

Odločite se za, ker je majhen, lep in prikupen, ker potrebujemo zvestega tovariša vedno v žepu, ker nismo preveč zahtevni in ker želimo za kakšen mesec razveseliti svoje otroke

Odločite se proti, ker je premajhen, da bi bil resen računalnik, ker cena ne ustreza vašemu žepu, ker sovražite računalnike ali pa boste kupili kaj večjega (upam da ne bom ob glavo) in resnejšega.

Tehnični podatki PC-1500A

Procesor CMOS 8-bitni
Pomnilnik 16 K ROm 8 5 K RAM
Jezik basic
Tipkovnica 65 tipk (alfanumerične funkcijske in programske)
Display 7×156 mini grafični LCD display, 26 znakov (velike in male črke, številke, posebni znaki itd.)
Zvok generator piribl 230–7000 Hz
Zunanji
Pomnilnik kasetofon programabilni moduli RAM
Vmesnik priključitev na CE 150

Tehnični podatki CE-150 Printer Cassette interface.

Sistem tiska/risanja pero (bal point), 4 barve
Širina papirja 58 mm
Hitrost pisanja 11 znakov na sekundo
Ločljivost 0 2 mm na korak
Znakov v vrstici standardno 18 (izbiramo lahko 36, 18, 12, 9, 7, 6, 5 in 4 znake)
Kasetni vmesnik 1200 bitov na sekundo

Cene v (DM) v konsignacijski prodaji Mercator-Contel

PC-1500A	*računalnik*	*440*
CE-150	*kasetni vmesnik in risalnik*	*420*
CE-152	*kasetofon*	*120*
CE-158	*centronics RS 232 C*	*390*
CE-161	*16 K RAM z baterijo*	*370*
CE-155	*8 K RAM*	*160*

Mavrični pogled na CBM – 64

Nadaljevanje s 5. strani

sploh kdaj napisali program, vreden uporabe) sledi 2 K pomnilnika za video (nizka ločljivost). Za basic je na voljo 38 K, nekaj malega manj torej kot pri mavrici 48 K. Basic ROM zavzame samo 8 K, sledi še 4 K RAM, idealno mesto za shranjevanje programov v strojnem jeziku. Čisto na vrhu je 8 K Kernalovega operacijskega sistema.

Opazili ste, da ni nikjer prostora za 8000 zlogov, ki bi jih porabila grafika visoke ločljivosti. Njeno mesto v pomnilniku je poljubno. Video čip mora samo vedeti, kje je začetek bitne karte. To mu povemo s pokanjem spominske lokacije v RAM. In če smo pametni, bomo grafiki določili mesto »pod« Kernalovim romom ali pod basicom. Slike pa bomo lahko na hitro menjavali samo s spreminjanjem sistemske spremenljivke in brez prenašanja gore informacij z enega na drugo mesto v pomnilniku. To je samo ena od značilnosti zares »pametne« strojne opreme.

Basic BASIC

Kratica BASIC v angleščini pomeni, da je to jezik za začetnike, pridevnik »basic« pa, da gre za nekaj najosnovnejšega, v prenesenem pomenu revnega. Torej si besedi v međnaslovu nasprotujete.

Od vse vdelane programske opreme je edino urejevalnik tak, da ga je vredno obdržati pri naslednjih verzijah Commodorovih strojev. Je namreč skoraj zaslonski, skoraj zato, ker je treba pri spreminjanju vrstic pritisniti na »ENTER«, da bi se popravek res poznal. V bistvu deluje ves zaslon kot orjaški vhodni pomnilnik (branje je zaradi podobne organiziranosti kot pri ZX 81 enostavno). Ko pritisnemo ENTER, računalnik analizira našo pozicijo na zaslonu in izvede ukaz tam, kjer smo bili. Vsi ukazi se tipkajo v celoti. Računalnik ob vnosu ne preverja sintakse. To je še en izvor napak pri izvajanju programa. Če naredi dosti tipkamo, si lahko pomagamo tudi s kombinacijami črk in grafičnih znakov. Na zaslonu potem namesto ukaza SAVE vidimo črko s in srček, kar ni pretirano pregledno.

Gorenjci in vsi, ki želijo pri CBM prihraniti nekaj prostora v pomnilniku, ne bodo pisali praznih presledkov za ukazi. Zato bodo izpisi navadnemu zemljanu še mnogo bolj tuji kot sicer. Vrstica programa je lahko dolga le 80 znakov. Umikanje ukazov v zankah, ki bi prav tako povečali preglednost, ni dovoljeno.

Beda

Commodorov basic je zgled vsega tistega, kar so ljudje očitali temu programskemu jeziku, ki je zaradi svoje okornosti in nerodnosti zgubil tudi svoj prvotni namen: pomagati začetniku, da bo napravil prve korake v svet računalnikov. Predstavljajte si zmedeni pogled novinca, ki bo brez težav vtipkal program, ta pa ne bo delal. Commodore se res lahko pohvali, da je njegov jezik standarden, to pa je tudi vse.

Basic razume tri tipke spremenljivk – celoštevilčne, znakovne in realne. Imena vseh so lahko dolga samo dva znaka. Polja so lahko eno-, dvo- ali tridimenzionalna, kar ustreza večini aplikacij. Ni pa mogoče velikosti polja ponovno dimenzionirati (DIM). Znakovne polje so enodimenzijska, dolga do 255 znakov, operacij z njimi pa je nekaj manj kot pri mavrici. Tudi pri funkcijah se pozna, da je roma le 8 K. Manjkajo ASIN, ACOS, s svojih pa tudi ne moremo definirati. Edina svetla točka so logične operacije, ki delujejo med biti. Te potrebujemo za krmiljenje zvoka in grafike skupaj z ukazi PEEK, POKE in USR. Posebnih ukazov, ki bi sicer krmilili skrite sposobnosti računalnika, sploh ni! Narisati črto je nemogoče. Načrtovalci so računali, da bodo uporabniki risali slike z množico grafičnih znakov, zaradi katerih so kode znakov čudne in na C-64 tudi ne moremo priključiti kateregakoli tiskalnika.

Med delom opozarja na napake blizu 30 sporočil, med njimi »FORMULA TOO COMPLEX« »preveč zapletena formula«. Zaradi takih in podobnih poenostavitev je izvajanje programa lahko res hitrejše kot pri spectrumu, vendar bolj zapleteni programi prav zaradi pomanjkanja ukazov trajajo dalj časa.

Programska oprema

C-64 je računalnik številka 4 v ZDA in to pomeni več, kot biti najbolje prodajan v Veliki Britaniji. Profesionalne tipkovnice zagotovi, da bodo računalnik kupovali ljudje, ki doma sestavljajo napišejo tudi kakšno pismo ali si izpišejo davčno napoved. Igralno usmerjena strojna oprema daje dovolj prostora programerjem iger. V primerjavi z mavrico je na voljo več in kvalitetnejše resne aplikativne programske opreme. Igre se bolje slišijo, toda kljub strojnim rešitvam so načelno nekaj malega počasnejše kot pri spectrumu. Programske opreme za programerje torej jeziki in uporabni programi, je še vedno nekoliko slabše kot pri spectrumu.

Ubožni basic je povzročil poplavo dodatnih jezikov. Nalaganje v RAM je zamudno, zato so najpopularnejše kartice ROM. Program Simon's Basic, ki vsaj delno odpravlje pomanjkljivosti basica, v romu smo predstavili v oktobrski številki Mojega mikra.

Kupiti ali na?

Če bi bil C-64 avto, bi bil namenjen predvsem tistim, ki ga potrebujejo za to, da se vsak dan vozijo v službo in občasno na krajši nedeljski izlet. Kaj se skriva pod pokrovom in kako iz stroja stisniti kaj več, jih ne zanima, saj spoštujejo omejitve hitrosti. Tudi perejo ga najraje v avtopralnici.

Commodore govori v reklamah o prijaznem računalniku, vendar to gotovo ni. Če bi se radi naučili osnov računalništva in programiranja, vam nakupa ne morem svetovati. Če pa nameravate uporabljati C-64 s kupljenimi programi, da vam bo pomagal pri delu, in občasno dovoliti mladeži, da sprosti agresijo v igrah, je to računalnik za vas. Za 700 DM boste dobili daleč najkvalitetnejšo strojno opremo. Sprašujemo se, ali ima Commodore kljub velikim serijam sploh kaj dobička s samimi računalniki. Priklenil pa vas bo nase z dodatki, o katerih govorimo v zadnjem delu revije.

Na legalnem jugoslovanskem trgu je C-64 zaenkrat daleč najboljši nakup za vsakogar, tudi zaradi siceršnje razširjenosti računalnika pri nas.

In končno, bi ga zamenjal za spectrum? Ne bi, saj me je ta vendarle navadil na nekaj stvari, ki jih pričakujem tudi od drugih mikroračunalnikov. Če bi imel CBM od vsega začetka, bi bile stvar podobne. Tudi CBM daje nevdušenim programerjem predvsem zaradi strojne opreme veliko možnosti. Le nekoliko »bolj iz nič« morejo začeti.

HVALIMO:

strojno opremo v celoti, posebej pa zvok in tipkovnico

GRAJAMO:

vdelano programsko opremo
nemogočo počasnost kasetofona in disketne enote
nestandardne priključke
nestandardni nabor znakov

ŽIGA TURK

Pozno pomladi je Commodore predstavil tri modele, ki naj bi osvežili njegovo ponudbo hišnih računalnikov. VIC 20 je postal vendarle preveč zastarel (a se je kljub temu še vse poletje odlično prodajal). Boj za tržišče je neusmiljen in uspeh modela 64 je Commodore spodbudil, da napade na vsej črti. Iz ene največjih (prodajnih) uspešnic v kratki zgodovini mikroračunalništva so razvili tri nove modele, ki jih je Moj mikro bežno predstavil že v svoji prvi številki.

Commodore 4+ (v začetku so omenjali tudi imeni 264 in 364) naj bi posegel po zgornjem delu trga osebnih računalnikov (personal computer), od koder naj bi izrinil BBC, memotech, IBM-PC junior in se postavil ob bok QL, vsaj dokler Commodore ne odgovori z resnično novim računalnikom. V bistvu je to C-64 z nekaterimi kozmetičnimi popravki in višjo ceno. Glavna cokla vzponu med računalnike novejše generacije je ozek zaslon (40 znakov). Zato je njegova uporabnost za poslovne aplikacije, katerim naj bi bil namenjen, avtomatsko sporna.

Commodore C-16 naj bi zasedel srednji del cenovne lestvice, C-116 pa pometel konkurenco v Spectrumovem razredu. Popolno obvladovanje trga torej, kot bomo videli s skoraj istim računalnikom.

C-116 ni stroj, ki bi mu kazalo posvečati posebno pozornost, toda zanimiv je prav zato, ker bomo ob njem spoznali lastnosti nove generacije commodorov. Žal smo ga od Konima dobili le nekaj dni pred zaključkom redakcije, zato test ne more biti popoln.

Pogled v tehnične podatke odkrije, da v seriji pravzaprav ni ničesar novega. Tu drugačna tipkovnica, drugje kak spominski čip več, pa kanal za generiranje zvoka manj. Kombinacij med tipkovnicami, pomnilnikom in zvokom je ogromno, pri Commodoru so se odločili za tri.

Druge karakteristike so tako rekoč identične kot pri C-64, le da so pri vseh uporabili nov video čip, ki ne zna generirati sličic (sprite), zato pa je 16 barv na voljo v 8 stopnjah svetlosti. ROM je enkrat daljši (32 K).

Basic na popravnem izpitu

Ime mu je BASIC 3.5. Na škatli za commodore 116 sicer piše, da gre za SUPER BASIC, a »super« je tu lahko kvečjemu pridevnik, ne pa podobnost s »Super Basicom« za QL. To sploh ne pomeni, da je

C-116: najšibkejši iz »nove« generacije

slab. Toda spet so leto ali dva za konkurenco. Nasploh jezik še najbolj spominja na oskubljeno varzijo basica MSX. Podobno kot pri računalnikih MSX je urejeno tudi premikanje kazalca s tipko za igralno palico.

Softversko je ostala grafika nespremenjena, vendar jo lahko zdaj vsaj uporabljamo. Grafika visoke ločljivosti pusti C-116 manj kot 2 K prostega pomnilnika, zato daljših umotvorov nisem mogel stlačiti v RAM. Mašanje teksta in grafike si tudi predstavljajo dokaj čudno: spodaj tekst, zgoraj grafika.

Ukazov za risanje je dovolj. C-116 zna nasti točke, daljice, kroge, elipse in kvadrate, zaprt lik tudi pobarva. V stavek za risanje lahko vključimo podatek o barvi. Specifična sta ukaza SCALE in SSHAPE/GSHAPE. Prvi ne dela tega, kar bi od njega pričakovali, ampak daje na izbiro tri različna razmerja med podano in narisano koordinato, kar naj bi pomagalo pri skakanju iz enega v drug grafičan način. SSHAPE in GSHAPE narišeta na poljubno mesto na zaslonu sličico, ki pa ni prava sličica, ampak niz znakov.

Še vedno je v naboru znakov precej za definiranih grafičnih znakov, ki so skoraj neuporabni.

Tudi oba kanala za zvok in šume sta programsko podprta. Po tem, kar piše v priročniku in kar sem slišal, se mi ni zdelo, da bi bil zvok tako dober kot v C-64. Uravnavamo lahko le višino, glasnost in trajanje zvoka ali šuma.

Tako risanje ni glasba. Toda tudi programiranje nasploh bo z novimi modeli lažje. Nekaj kontrolnih struktur je povzetih po strukturiranih jezikih (DO LOOP, WHILE, UNTIL, IF, ELSE), mogoče pa je definirati tudi vrstične funkcije. Ne boste verjeli, toda GOTO zna skočiti v vrstico, ki je ni, in RENUMBER ga zna preštevilčiti.

Uporabnost vseh naštetih dobrot pa ni zanesljiva, saj nas spet omejuje dolžina imen spremenljivk: dva znaka. Pisanje bolj zapletenih programov bo težavno, še posebej s C 4/+, ki ima nekaj več pomnilnika.

Cenena plastika z znano etiketo

Basic 3.5 ima vsa letošnja serija commodorov. Nimajo pa vsi, hvalabogu, gumijaste tipkovnice. Ookler nisem položil prstov nanjo, sem mislil, da ima spectrum najslabšo »gibljivo« tipkovnico. Še nikoli v življenju se nisem tako mučil pri tipkanju, niti z ZX 81 ne. Na dolgo tipko za presledek je sploh treba mahati s kladivom... Škatla je taka, kot da so znotraj lučke za novoletno jelko. Računalnik je plastičen, a s tipko RESET in ON/OFF ter s priključki, ki jih ima Commodore (združljivost!).

Commodore, serija 1984
Tehnični podatki

	C-64	C 116	C 16	C4/+
Procesor	6510	7501	7501	7501
RAM	64	16	16	64
RAM basic	38	12	12	60
RAM-mc	56	12	122	60
ROM	16	32	32	32
Grafika	320×200	320×200	320×200	320×200
Tekst	40×25	40×25	40×25	40×25
Zvok	2+1	1+1	1+1	2+1
Software	BASIC	BASIC&mntr	BASIC*mntr	BASIC&mntr&
Tipkovnica	71 tipk	71 radirk	71 tipk	71 tipk
Cena (DM)	886	416	459	1250
+55% dinarskih stroškov				

Kupiti ali ne?

Spectrum 16 K ponuja objektivno več kot C-116, pa še razširite ga lahko. Commodore morda upa, da bo kdo vzdihnil ob imenu računalnika in ga kupil. Zato tovarno, ki je znana po dobrih tipkovnicah na poceni računalniku, je C-116 sramota. Odgovor je torej: ne kupiti, predvsem uvozniku, ker tihotapci že sami premislijo, kaj se splača in kaj ne. Morda je za Jugoslovane, ob napovedani ceni 1250 DM za 4/+, ta hip še najboljša izbira C-16, seveda z razširitvijo pomnilnika.

Sinclair QL: skok s prestopom? (2)

CIRIL KRAŠEVEC
ŽIGA TURK

V nadaljevanju našega superteста si bomo pobliže ogledali strojno opremo, procesor MC 68008 in programe, ki so za ta računalnik že na voljo.

Strojna oprema

Največji revezi med hackerji so komodorjevci. Trudijo se s tremi 8-bitnimi registri in kupom posebnosti glede uporabe prve strani spomina. Pri ZX spectrumu smo nekoliko na boljšem, registrov je nekaj več in tudi kakšno 16-bitno število znajo prezvečiti. Programiranje v strojnem jeziku s QL pa bo, če se izrazim avtomobilsko, kot vožnja s ferrarijem proti trkljanju s fički in katrami.

Najprej enkrat za vselej razčistimo s tem, koliko bitni je mikroprocesor MC 68008. Tisti, ki se na spoznajo in si radi privoščijo »strička Clivea«, so mu hitro pripisali potegavščino, češ da MC 68008 sploh ni 32-bitni procesor, tako da bodo ubogi lastniki spet začeli dobivati manjvrednostne komplekse. »Bitnost« procesorja določajo trije podatki: širina podatka, ki ga procesor »naenkrat obdela« (notranja arhitektura), širine podatkovnega vodila, torej količine informacije, ki jo procesor naenkrat sprejme ali odda, in širina naslovnega vodila, ki določa, koliko pomnilnika lahko procesor naslovi. MC 68008 ima 32-bitno notranjo arhitekturo, vsaj v pomenu, v kakršnem sodi 6502 med 8-bitne procesorje (ne dela pa ničesar ne 64 bitih, kot počne npr. Z 80 na 16). Ima 20 naslavljalnih nožic, torej lahko naslovi 1 048 576 zlogov, 16-krat manj kot drugi procesorji iz serije 68000. Edina pomembna razlika pri verziji 68008 tega procesorja je samo 8-bitno podatkovno vodilo, kar je natanko toliko kot pri Z 80. Motorola imenuje model 68000 16-bitni mikroprocesor in mislim, da ustreza taka oznaka tudi modelu 68008. Po mnenju mnogih so procesorji serije 68000 najmočnejši med vsemi 16-bitnimi mikroprocesorji!

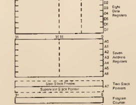
MC 68008

Mikroprocesor MC 68008 je razvila Motorola, poleg nje pa ga izdelujejo Mostek Corp., Hitachi Ltd., Philips, Rockwell Int., Signetics in Thomson EFCIS. Takšna razvejenost proizvajalcev je zelo pomembna za uporabnika, kakršen je Sinclair: oskrbovanje z osnovnimi elementi iz več virov zmanjšuje možnost, da bi nastal v proizvodnji izpad zaradi industrijskih ali političnih igric.

Kot smo že povedali, je MC 68008 šestnajstbitni mikroprocesor z osembitnim podatkovnim vodilom. Takšno vodilo poceni in poenostavi uporabo procesorja v konfiguracijah z osembitnimi periferními vezji. To je tudi vzrok za nizko ceno QL. Vsi mikroprocesorji iz generacije 68000 imajo enak nabor ukazov, tako da program, pisan za model 68000, dela tudi v modelu 68008.

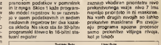
Ker 68008 nima posebnih vhodno-izhodnih instrukcij, je programiranje manj zapleteno kot z Z 80. Pri tolikšnem obsegu pomnilnika si lahko privoščimo, da nekaj naslovov predpišemo kar periferním enotam, pomnilniško celico in periferijo dekodiramo ali določimo z isto instrukcijo.

Znotraj 68008 se informacija prestavlja med 17 večnamenskimi registri. Vse aritmetične in logične operacije se lahko izvajajo v vseh registrih. Rezultati se lahko prav tako vpisujejo v isti register, s tem da se vse zastavice (flags) v statusnem registru postavijo v ustrezno stanje.



Programski model registrov

Veliko število 32-bitnih registrov nam da vedeti, da se zelo zahtevne operacije lahko izvajajo znotraj procesorja z minimalnim prenosom podatkov v pomnilnik in iz njega. Skica 1 kaže programski model registrov, ki se razvrstijo v osem podatkovnih in sedam naslovnih registrov ter dva kazalca sklada. Na skici sta še 32-bitni programski števec in 16-bitni statusni register.



Shema statusnega registra

Statusni register (skica 2) je razdeljen v dva byta. Uporabniški byte sestavljajo običajne zastavice in razširjen nabor zastavic za označevanje izposoje ali prenosa v računanju z več zaporednimi dolgimi besedami. Razširjen nabor zastavic je zelo podoben zastavici »prenos« (carry) pri drugih mikroprocesorjih.

Drugi del statusnega registra je sistemski byte. Sestavljajo ga prekinitve maske. S tremi biti določimo prioriteto prekinitve. 68008 ima v nasprotju z 68000, kjer so trije prekinitveni vhodi, samo dva. Vhoda IPL0 in IPL« lahko kombiniramo tako, da podpiramo tri prekinitvene nivoje 2, 5 in 7. Prekinitvene nožice mikroprocesorja zaznajo vkodiran prioritetni nivo prekinitvenega vezja, nivo 7 ima najvišjo prioriteto in ni maskiran. Na vseh drugih nivojih so lahko prekinitve maskirane. Pri izvajanju programa spozna procesor samo prekinitve višjega nivoja, kot je tekoči.

Podobno delajo ukazi reset, trace, software restart in trap.

Zelo pomembna lastnost 68008 je, da je ustrezen za delo v večprocesorskih vezjih. Sinclairovi načrtovalci so naredili dobro delo, ko so prepustili nekaj sistemskih funkcij drugemu procesorju (8064). Ta lahko zahteva uporabo vodila in signalnih linij mikroračunalnika s signalom »bus request« nožica (BR). 68008 takoj po sprejemu te zahteve postavi naslovne, podatkovne in kontrolne izhodne linije v stanje visoke impedance, tako da lahko drugi procesor prenaša podatke v pomnilniške celice in iz njih (ali pa iz periferije).

Mikroprocesor MC 68008 ima tudi kontrolne signale za povezavo s periferními vezji osembitne generacije 6800. Ti signali so potrebni zato, ker operira 68008 s podatki asinhrono, 6800 pa sinhrono.

Strojni jezik

MC 68008 ni bil razvit za to, da bi manjšim in večjim otrokom omogočal uničevati vesoljske napadalce. Načrtovalci v Silicijevi dolini so imeli v mislih mikroprocesor, ki bo srce miniračunalnika, bodisi sam ali v povezavi z drugimi procesorji. Velikanska razlika med mini in mikro pa je zahteve, naj lahko miniračunalnik uporablja več ljudi hkrati, mikroračunalnik pa en sam človek. Zbirnik in notranja arhitektura procesorja imata zato bogat nabor instrukcij, ki določajo prioriteto posameznim opravilom in nadzirajo njihovo delovanje, torej močno poma-

gajo operacijskim sistemom. Več o tem pozneje.

Izdelali so natančne študije o tem, kako pogosto se uporabljajo posamezni strojni ukazi. Dosegli naj bi, da bi bili najvažnejši med njimi kar najhitrejši in najkrajši. Mislili so tudi na čim lažje prevajanje višjih strukturiranih programskih jezikov, kar kažejo ukazi za delo s skladi in čakalnimi vrstami.

Zbirnik za MC 68000 (ali 68008), ima 56 osnovnih ukazov. Če bi upoštevali še vse njihove različice z različnimi operatorji, bi našteli nekaj desel tisoč kombinacij. Pohvalna lastnost procesorja je, da lahko vse počnemo z vsemi registri. Nobedan ni posebej povzdignjen v stan akumulatorja in nima kakšnih posebnih lastnosti kot HL v Z 80. To programerju močno olajša delo. Veliko število ukazov pa žal pomeni, potrebujemo kar nekaj bitov, da jih med seboj ločimo. MC 68008 v bistvu organizira pomnilnik v besede, dolge 16 bitov. Tudi najpreprostejši ukazi strojne kode tako zasedajo dva zloga. Ko smo primerjali, koliko je pri QL spravljeno v 48 K ROM in koliko pri spectrumu v 24 K, smo se strinjali, da je koda pri MC 68000 za 10–20 odstotkov daljša od ustrezne pri Z 80.

Podatki, na katere se naneša katerikoli ukaz, so lahko različno dolgi. Najkrajša je žarnica (bit). Sledijo 4-bitna BCD cifra, 8-bitni zlog (byte), 16-bitna beseda (word) ali 32-bitna dolga beseda (long word). Procesor pozna tri glavne načine navajanja podatkov:

– z registrom
– z efektivnim naslovom
– z nekaterimi ukazi za uporabo natanko določenega registra

Registrov je skupaj 19. Razen statusnega (podoben je registru zastavica v Z 80) so vsi 32 bitni.

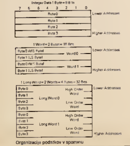
DØ do D7 so podatkovni registri. Izmed vseh so najobičajnejši in ne morejo kazati na zunanje pomnilniške lokacije, npr. (HL).

Prvih šest naslovnih registrov (AØ do A6) lahko uporabljamo, kot da gre za navaden register »D« ali pa na pet načinov kot kazalec na pomnilnik:

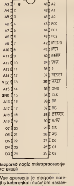
– indirektno (efektivni naslov je zapisan v nekem registru »A«)
– indirektno s povečanjem kazalca po operaciji (podobno vplivu ukaza POP GET na kazalec na sklad – SP)
– indirektno z zmenjšanjem kazalca (PUSH)
– indirektno z razliko (efektivni naslov je vsota registra in 16-bitnega števila – poenostavi pisanje premakljivih programov)
– indirektno z indeksov (efektivni naslov je vsota 8-bitnega števila) nekega drugega (indeksnega) registra.

Vsega skupaj je 14 načinov naslavljanja.

Vsak naslovni register lahko uporabimo kot kazalec na sklad, indeksni register ali kazalec z razmikom. Prav vloga kazalca na sklad omogoča programerju, da v RAM ustvari več skladov (LIFO – last in first out) ali pa z dvema kazalcema naredi VRSTO, torej uredi na način FIFO (first in first out).



Organizacija podatkov v spominu



Razpored nogic mikroprocesorja MC 68008

Vse operacije je mogoče narediti s katerimkoli načinom naslavljanja, torej z enim ukazom vzamemo kakšno število iz sklada, premaknemo kazalec in število kam prištejemo.

Med ukazi za prenos podatkov naj omenimo samo LINK, ki porine v sklad vrednost naslovnega registra. V naslovni register se zapiše kazalec na sklad in končno se kazalec na sklad poveča za poljubno 16-bitno razliko. Ukaz UNLINK vrne v SP vrednost, ki je v nekem naslovnem registru. Slednjemu se nato priradi iz sklada vzeta vrednost. Kratek premislek pokaže, da lahko našteta ukaza uporabimo za vzdrževanje povezanih podatkovnih struktur (linked list) in področij parametrov v skladu, ki se bodo učinkovito prenašali tudi v podprograme.

Tok programa spreminjamo z relativnimi in absolutnimi skoki ter s klici podprogramov. Dodamo lahko tudi pogoje, ki jih je 16, spet več kot pri drugih mikroprocesorjih. Mnogi med ukazi programerju res kaj pomenijo, npr. LE (manjše ali enako), NE (neenako). Edino zastavico CARRY v statusnem registru preizkušamo, kot smo navajeni z 8-bitniki.

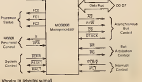
Relativni skoki so lahko do 32 K daleč, torej je lahko razlika 16 bitno število. Brez težav je mogoče zapisati ves program relativno. Kam skočiti, lahko povemo tudi z enim od načinov za navajanje efektivnega naslova.

Petnajst ukazov je za celoštevilsko aritmetiko. Poleg seštevanja in odštevanja zna procesor množiti in deliti! Zato je posebno privlačen za aplikacije, kjer se veliko računa. Logične operacije omogočajo AND, OR, EOR (XOR) in NOT pri katerikoli velikosti podatka, podobno kot pomikanje in vrtenje.

Največjo moč nad procesorjem pa tudi nad programi v romu QL dajejo ukazi, ki kontrolirajo delo procesorja. Pri Z 80 imamo v ta namen pravzaprav samo INTERRUPT. MC 68000 je vedno v enem izmed treh stanj: normalnem, izjemnem (exceptional) in ustavljenem.

Normalno stanje pomeni branje ukaza, ki ga kaže PC, in njegovo izvajanje. Posebno stanje je povezano s prekinitvami (interrupt) ter ukazoma TRAP in TRACE. Posebno stanje lahko dosežemo z ukazom ali neobičajnim stanjem v procesorju. »Izjeme« so namenjene obvladovanju nenavadnih in prepovedanih stanj znotraj procesorja.

Med delom je procesor bodisi v uporabnikovem ali v nadzornem stanju. Stanje določa procesorju, katere operacije so legalne. V ta namen sta tudi dve kazalca na sklad. Ta zasnova omogoča npr. kontrolo dela več uporabnikov hkrati: napaka v programu enega ▶



Vhodni in izhodni signali

uporabnika ne more škodovati drugemu saj ne more dosuči drugih pomnilniških lokacij razen svoje

Zanimiv je tudi ukaz TRACE ki omogoča da sladimo poteku strojnih programov Po vsakem ukazu naredi procesor izjemo v podprogramu ki npr izpiše ukaz in čaka na pritisk na tipko

Natančen opis teh »eksotičnih« posebnosti MC 68000 presega okvir našega supertesta Posvetili se jim bomo kdaj pozneje, ko bo QL bolj razširjen

Dokler opazujemo QL takega kot smo ga dobili na vamo, da je to la vrh ledene gore ki ga je morda mogoče primerjati s C 4 plus ali kakšnim drugim strojčkom Razlika med tem, kar računalnik tipa QL lahko naredi, in kar bo še zmogel, je ta hip približno taka kot med ZX 81 z 1 K in spectrumom z 48 K Prav to so pri mnogih testih premalo poudarili

Zbirnika za QL

Že v reportaži iz Londona (MM oktober) smo poročeli, da naj bi bila za QL na voljo dva zbirnika Naši sodelavci so naročili oba, podjetji sta hvaležno sprejeli denar programov pa od nikoder Upamo, da ti zbirniki niso tako visoka tehnologija da je ne bi smeli uvažati v Jugoslavijo Oceno povzemamo po prospektih obeh proizvajalcev in oceni v reviji Electronics & Computer Monthly

Prvi zbirnik so napisali v hiši Metacomco sicer znani po softvarskih projektih za Motoroline mikroprocesorje Za QL so predstavili tudi alternativni operacijski sistem, 68 K/OS ki naj bi imel pred QDOS nekaj pomembnih prednosti

Zbirnik za MC 68000 so prepisali iz drugega, mnogo večjega računalnika To kaže da programska oprema za QL še zdaleč ne bo vezana na umotvore hackerjev Zbirnik je dolg celih 60 K kar pomeni, da je za izvorno in prevedeno kodo na voljo skupaj okrog 30 K Ta popolnoma profesionalni izdelek ne prevaja neposredno v RAM pač pa v datoteko za katero po slabih izkušnjah iz Psionovih programov upamo da se da shraniti tudi v RAM Zbirnik ima vdelan svoj zaslonski urejevalnik verzijo priznanega programa dr Tima Kinga ki se v razvojnem sistemu BCPL ukvarja tudi z BBC microm Hitro delovanje tega urejevalnika naj bi »enkrat za vselej spravil Psion v zadrego« Avtor ga priporoča za urejanje česarkoli, pa naj bo to tekst, program v Super Basicu ali pismo prijatelju Znotraj urejevalnika je mogoče s premikanjem kazalca določati velikost okan, črk itn Seveda so vdelani vsi standardni dodatki iskanja teksta, zamenjava vstavljanja premiki blokov

Koda, ki jo zbirnik generira je lahko absolutna neodvisna od pozicije ali premakljiva. Glede na zahteve QDOS je najbolj priporočljivo uporabljati zadnjo možnost

Program seveda dopušča makro ukaze in knjižnice Navodila za uporabo obsegajo dve knjižici, eno za urejevalnik in eno za zbirnik za mikrotračnik pa menda dobita tudi datoteko z vsemi definicijami za stavke QDOS in EQU za klice procedur v ROM

Drugi zbirnik so napisali pri CST in ga pozneje prodali Sinclairu, kjer naj bi vsak hip izšel z njihovo nalepko Metacomcov postavlja standard za kvaliteten zbirnik in podobno kot Psionovi programi zahteva od proizvajalca da poveča zmogljivost strojne opreme CST in posredno Sinclair se strojni opremi podrejata s 17 K dolgim zbirnikom ki seveda more zaostajati za konkurenčnim izdelkom Nekaj prostora pa so gotovo pridobili z več klici v ROM Prostor bo prišel posebej prav vsem, ki bodo želeli imeti pri delu tišino in ne bodo trpeli da bi se vsake toliko oglasili mikrotračniki Prevajali bodo lahko v RAM, kot so navajeni pri manjših računalnikih

Za malo denarja malo muzike Sinclairov zbirnik stane 40 Metacomcov pa 60 funtov Cene softvara torej naraščajo s ceno strojne opreme Sinclairov zbirnik lahko naročite na običajni naslov v Camberleyu, kvalitetnejšega pa dobite pri družbi Metacomco, 26 Portland Square, Bristol, Great Britain

Quill: časovna leča

ŽIGA TURK

Zagrizeni računalnikarji v svoji profesionalni deformiranosti pogosto pozabljamo da večina sploh ne namerava programirati računalnikov, ampak bo uporabljala že gotove programe Pri Sinclairu so pomislili tudi na to in dodali računalniku QL štiri programe ki jih potrebuje vsak uporabnik mikroračunalnika, ne glede na stroko

Veliko člankov, ki jih objavlja Moj mikro, ne nastaja na pisalnih strojih ampak si avtorji pomagamo z računalniki Tale tekst je prvi resni avtorjev poskus s programom QL

Pred samim pisanjem je zaslon razdeljen na tri področja Ker sem ob vklopu računalnika izbral televizijsko velikost oken, imam na zaslonu 25 vrstic s 64 stolpci Torej bom lahko tudi na zaslon pisal tekste tako na široko kot jih bom pozneje izpisal na papir Širina zaslona je bistveno večja kot pri nekaterih drugih računalnikih ki naj bi QL neposredno konkurirali Če je na zaslonu slaba slika si lahko z ukazom DESIGN izberem tudi manj pa zato večjih znakov v vrstici (40) Število znakov ki jih vidim na zaslonu, načelno nima nobene zveze z dejansko širino teksta Če je tekst širši od širine zaslona, se ta začne pomikati vstran Zaslon je videti zelo profesionalen zelene črke na črni podlagi

Skupna značilnost vseh Psionovih programov je izredna prijaznost do uporabnika. V zgornjih štirih vrsticah mi program v vsakem trenutku sporoča, kaj lahko naredim Če sem v zadregi lahko s pritiskom na F1 – HELP zaprosim za pomoč Navadno si mora program pomagati z branjem mikrokasete potem pa mi da nekaj nasvetov v zvezi z operacijo, pri kateri sem zabredel Spodnji vrstici mi posredujeta vse podatke o mojem delu npr število besed vrstico na strani, stran, naslov teksta Pozicijo kazalca označuje tudi svetla pika na ravnilu v peti vrstici

Osnovna naloga urejevalnika tekstov je omogočiti uporabniku da bo s kar najmanj truda vstavljal besedilo ga popravljal in v lepi obliki prenašal na papir Tekst se pojavlja na mestu, kjer je kurzor Ta ni utripajoč kar gotovo pomirjevalno vpliva na uporabnika Tekst lahko vstavljam v dveh načinih INSERT ali OVERWRITE Pri prvem programu vriva tekst Če npr izpustim črko se s kazalčnimi tipkami premaknem na mesto kjer manjka in jo vtipkam Program bo zanjo samodejno naredil prostor in poravnal novo obliko teksta tako kot bom določil z ukazom JUSTIFY Med vnašanjem teksta mi ni treba paziti, kdaj bom prišel do konca vrstice Program že med pisanjem poravnava tekst in če bi moral besedo deliti celo premakne v novo vrsto Med besede stare vrstice vstavi toliko presledkov da je desni rob poravnan Če uporabljamo daljše besede, je lahko na ta račun med njimi mnogo presledkov Pri Quillu se da določiti kje sme program besedo deliti Po svoje rešitev ni najboljše, saj beseda ki smo ji pravkar določili, kako se deli, po urejanju teksta mroda ne bo več na takem mestu, da bi jo bilo treba deliti Po delitvi se besedi tudi nikjer ne pozna da smo ji že določili kako se deli

To gre na račun filozofije »What you see is what you get« (Kar vidiš na zaslonu bo izpisano na papirju) Quill vas med izpisovanjem teksta na tiskalnik nikoli ne pusti v negotovosti Natanko tak bo kot ste ga videli na zaslonu To pride posebej do izraza, kedar želimo kakšen tekst zapisati krepko, ga podčrtati ali pa napisati formulo z indeksi in eksponenti $E=MC^2$ je na zaslonu natanko tak kot na papirju Poleg večje preglednosti to izključuje možnost napak ki so se pogosto dogajale pri delu z drugimi urejevalniki tekstov (pozabljen znak za konec podčrtovanja je imel za posledico, da je bil ves tekst podčrtan) Tip črk spreminjam s tipko F 4 potem pa z eno od črk, ki označuje način pisanja

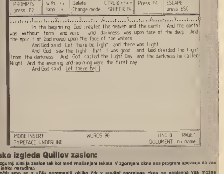
S pritiskom na F 3 pridem v t.i. COMMAND MODE, ukazni način

V zgornjem oknu mi program izpiše vse ukaze, ki so na voljo. Ti mi omogočijo shranjevanje datoteka na mikrotračno enoto, dodajanja drugih datotek ali rezultatov drugih programov na mesto kurzorja. Z njimi lahko oblikujem izpisane strani, torej število vrstic na strani, način oštevilčenja strani (rimske, arabske številke ali črke), definiranje glava (teksta, ki se bo izpisal na vrhu vsake strani), »repa« (footarja), ki se bo izpisal na koncu, pa seveda še vse ukaze, ki so pri boljših urejevalnikih besedil normalni: iskanja teksta in zamenjava s kakšnim drugim, premik delov tekstov na drugo mesto v datoteki, definiranja tabulatorjev, ki znajo avtomatsko poravnavati.

Vse ukaze spremlja pojasnilo oziroma menu, če je možnosti za izbiro več. Žal ima ta prijaznost do uporabnika tudi slabe strani. Program je na trenutke nemogoče počasen. Brisanje črk je na račun sprotnega poravnavanja vsega teksta pod njimi še najbolj podobno počasnemu posnetku istega opravila pri urejevalniku INES. Nekaj hitreje gre, če brišemo besede ali kar cele vrstice, vendar moramo v ta namen naštudirati vrsto kombinacij SHIFT, CONTROL in kazalčnih tipk. Priznati pa je treba, da prav noben pritisk na tipko ne gre v nič in da kljub pogostemu oglašanju mikrotračnika tipkarju ni potrebno čakati. Ima pa slabo lastnost, ki jo poznajo vsi, ki delajo s kakšnim preobremenjenim velikim sistemom. Odgovor na tipko pride šele po nekaj desetinkah sekunde. Posebej napeto postane prav pri brisanju, ko se kazalec noče ustaviti na točki, kjer bi želeli. Tudi ukazi, ki se ne nanašajo neposredno na tekst, se zaradi poseganja v mikrotračnik ne izvajajo dovolj hitro.

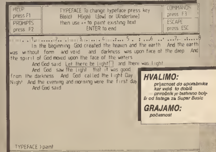
Teksta je v računalniku naenkrat zelo malo. Sprašujemo se, ali vsa ta prijaznost odtehta počasnost programa. Obdelovanje teksta je gotovo delo, ki zahteva od vseh računalniških aplikacij največja upoštevanje ergonomskih zakonitosti. Program sicer poskuša izvajati nekatera opravila takrat, ko ima občutek, da ga že nekaj časa nismo potrebovali. Kaže, da je PSION znotraj tega koncepta in načina programiranja (menda so uporabili C) naredil vse, kar je bilo v njegovi moči, toda to očitno ni bilo dovolj.

Če bi Quill tekel nekoliko hitreje, bi bil enakovreden najboljšim urejevalnikom besedil za »prave« poslovne računalnike, kot so IBM PC ali LISA. Tisti, ki bodo obdelovalnik zares pogosto uporabljali, se bodo verjetno odločili za verzijo programa v romu, ki bo zmanjšala posege v mikrotračnik in sprostila RAM za daljše tekste.



Tako izgleda Quillov zaslon:

Na zgornji sliki je zaslon tak kot med vnašanjem teksta. V zgornjem oknu nas program opozarja na vse kar lahko naredimo.
Odločili smo se z »F4« spremeniti obliko črk v sredini zgornjega okna se napisane vse možne kombinacije.



HVALIMO:

prijaznost do uporabnika
kar vidiš, to dobiš
priročnik je bistveno boljši od tistega za Super Basic

GRAJAMO:

počasnost

PETER MIRKOVIČ

Računalnik je postal kartograf

Brez grafično prikazanih podatkov si poslovneži v svetu niti zamisliti ne morejo preglednega in učinkovitega odločanja. Kajti v množici informacij in številk v tabelah tudi zelo bistra glava ne more hitro »izluščiti« potrebnega sporočila. Konkurent zamudo kaznuje. Zmotno pa bi bilo trditi, da se pomena grafično prikazanih podatkov pri nas ne zavedamo. čedalje več takšnih podatkov je tudi v delegatskih gradivih. Zato Računalniški atlas občina Sežana, predstavljen pred dnevi v Ljubljani, na bi zbudil zanimanja, če tematskih kart v atlasu jih je 40, v celoti ne bi izrisal – računalnik.

Gre za prvi tovrstni atlas v Jugoslaviji in prvi del projekta, imenovanega **Teritorializacija podatkov**, projekta z veliko večjimi ambicijami. Ze prihodnje leto naj bi namreč bila nared računalniška atlasa Slovenija in Jugoslavija. V teh atlasih bo mogoče zelo pregledno primerjati razvoj posameznih območij v najpomembnejših dejavnostih družbenega in gospodarskega življenja.

Prednosti računalniška izdelave kar pred klasično so zelo velike. Računalniška izdelava je zlasti hitrejša in natančnejša, pa tudi sorazmerno cenejša je. Medtem ko kartograf za klasično risanje karte potrebuje več ur ali dni, jo avtomatski risalnik izdela v ani sami minuti. Pregledno in brez napak, če so pripravljeni podatki dobri.

Razlog, da je bil prvi računalniški atlas pripravljen v Sloveniji, je preprost: le ta republika ima zakon, ki zahteva register teritorialnih enot in evidenco hišnih številk. V drugih republikah zavodi za statistiko teh podatkov ne zbirajo. In zakaj atlas Sežane? To občino ob meji so snovalci izbrali zato, ker je v mnogih pogledih tipična slovenska občina.

»Računalniški atlas sežanske občine naj bi pravzaprav opozoril na novo tehnologijo, s katero v avtomatizirani kartografiji izdelujemo izvirne tematske karte za večbarvni tisk in pri tem uporabljamo pogojne kartografske znake. Vsebina atlasa je bila zato za zdaj manj pomembna,« pravi **Janka Rozman** z ljubljanskega Inštituta za geodezijo in fotogrametrijo, raziskovalec, ki je pripravil programsko opremo za ta atlas. Na ta način so nameravali nosilci projekta (glavni so slovenski zavod za statistiko, slovenska geodetska uprava in geodetski zavod, nekateri inštituti in slovenski zavod za družbeno planiranje) pridobiti programsko opremo za serijsko izdelovanje podobnih tematskih kart. To se jim je v veliki meri tudi posrečilo. S teritorializacijo podatkov pa naj bi opozorili najrazličnejše organizacije in službe, ki zbirajo in obdelujejo podatke, kako pomembna je prostorska opredelitev podatkov. Kajti za izdelavo tematskih kart je treba zadostiti naslednjim pogojem: da so na voljo zanesljivi podatki, strojna oprema, programska oprema in ... denar.

input: natančni prostorsko opredeljeni podatki

Kakor pravi raziskovalec Rozman, brez prostorsko opredeljenih podatkov ni mogoče izdelati niti klasične tematske karte. Tudi pri računalniški morajo biti vsi podatki locirani v prostor bodisi prek **resnične lokacije** ali prek **prostorske enote.** Pri resnični lokaciji je delo zahtevno, določiti je treba koordinate na zemljišču ali

na podlagi podatka vrisanega na karti. Lociranje je lažje z uporabo prostorske enote. Z mrežo prostorskih enot vrisanih na karti je mogoče določiti prostorsko pripadnost, kot pravijo kartografi. Vse podatke, ki so zbrani za isto prostorsko enoto, lahko med seboj primerjajo ali celo združujejo. Tudi zaradi teh potreb je Slovenija uvedla zahtevo po ROTE (register območij teritorialnih enot) in EHIŠ (evidenca hišnih številk), oboje pa pomeni mrežo med seboj povezanih prostorskih enot. Vse prostorske enote so natančno locirane z zarisom v geodetskih načrtih in kartah. Za nekatere občine in prostorske enote so določene tudi koordinate obrisov – pri čemer so anologni grafični podatki spremenjeni v digitalne – in koordinate **centroidov**, značilnih točk prostorskih enot, ki so običajno v težiščih. Tako ROTE in EHIŠ olajšujeta izdelovanje računalniških tematskih kart, toda pod pogojem, da so podatki opredeljeni po prostorskih enotah. Vedno ni tako. Kartografi pravijo, da jih statistični zavodi oskrbujejo z natančnimi ekonomskimi in socialnimi podatki. Zapleta pa se pri zbiranju prostorskih podatkov, denimo takrat, ko določajo pogozdenost območij. Pomagati si morajo z mehaničnim (zamudnim) merjenjem, če ni na voljo zemljevidov, in s fotografskimi posnetki iz letal.

Ker »računalnik ne dela napak«, so podatki o prostorskih enotah oziroma koordinatah bistven pogoj za uporabo avtomatizirane kartografije. Digitalne podatke je mogoče pridobivati iz grafičnih analogno na več načinov. Ročno zajemanje bi bilo preveč zamudno, zato jih zajemajo z digitalizacijo ali s **skanerjem**, (vzorčnikom), napravo, ki po pasovih registrira na monitorju (piksle) in meri refleksivno valovno dolžino. Izhod so nizi koordinat ali matrike podatkov na majhnih kvadratih, naprave jih loči glede na intenzivnost (na zemljevidu so črne meje, prav tako objekti ali tiri, s programom pa jih loči po linijah).

Ko so digitalni podatki prostorskih enot in drugi podatki na voljo, risalnik izdela sliko na podlagi numeričnih podatkov. Prvi so se pojavili inkrementalni risalniki, ki so risali le v določenih smereh. Pozneje so na Zahodu za natančnejše risanje razvili numerično vodene koordinatografe, ki omogočajo risanje tekočih krivulj, graviranje in risanje s svetlobnim snopom (laserjem). Med zadnjimi so se pojavili **rastrski risalniki**, s katerimi so pripravili tudi računalniški atlas sežanske občine. Osnovni element takšne slike je pika, hkrati eden od elementov matrike podatkov. Od gostote pik je odvisna natančnost risanja.

Vse tematske karte omenjenega atlasa so izdelane z računalnikom DEC-10, zrisala pa sta jih risalnika versatec, na katerima slika nastaja po celotni širini hkrati, večji risalnik zriše slike do širine skoraj 90 centimetrov, dolžina slike je neomejena, v sekundi pač zriše 2 centimetra slike v vsej širini. Računalniški sistem sodi v zgornjo polovico srednje velikih sistemov, natančnost elektrostatičnega rastrskega risalnika pa je 0,12 milimetrov, kar zadošča za tematske karte. To potrjuje dejstvo, da je bila večina tematskih kart izrisana v merilu 1:1.

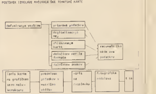
Iz šestih rastrov nastana paleta 343 barv

Med programsko opremo, ki jo je za izdelavo tematskih kart sežanskega atlasa prispeval Janko Rozman, sodijo tri glavne skupine: programi za pripravo podatkov, za pripravo in korekcijo slik ter za izdelavo slik na risalniku. Pomemben del so datoteke s kodami rastrov, vzorcev in znakov. Izvajalci projekta imajo tačas na voljo 260 rastrov in vzorcev ter 200 kartografskih tematskih znakov, tudi precej zahtevnejših, kot si jih lahko privoščijo v klasični kartografiji. Iz šestih posebnih pikčastih rastrov lahko pridobijo s pretisi barvno paleto z nič manj kot 343 barvami.

Kartografska podatka bi lahko pripravili s spactrumom

»Programi so pripravljeni v modularni obliki,« razlaga Janko Rozman, »modul je samostojni del, ki lahko zriše svoj program, vender univerzalnega programa, takšnega, ki bi bil uporaben za več deset občin, ni. Podatki v drugih občinah niso prilagojeni tako kot v sežanski, zato je treba spreminjati module in spremembam prilagajati programske aplikacije.«

Rozman je pojasnil tudi računalniško obdelavo. Pri tej je bila kot pravi, včasih potrebna predelava podatkov, ko so pripravljali sintezne karte, na katerih so bili grafični prikazi več območij. »V tej fazi izdelamo prve izrise na grafičnem terminalu. To so kontrolni izrisi, ki omogočajo korekcijo risalnih centroidov in velikost enote risanja znakov ali diagramov. Ko je karta pripravljena za izris na risalniku, sledijo štiri faze izdelave slika na risalniku. Ker je



večina podatkov v vektorski obliki, dobimo najprej absolutno orientirane vektorje. To dobimo z aplikativnim uporabniškim programom, ki kliče različne sistemske in uporabniške grafične module. V drugi fazi predelamo vektorje z urejevalnim programom v relativne vektorje, ki pomenijo rastrsko sliko in so vhod na risalnik versatec. Te podatke je treba zapisati na trak in ta trak prenesti na risalnik, potem pa čitalna enota bere rastrsko zapisani trak in ga pošilja v krmilni procesor, ki posreduje signale risalniku. Če izdelujemo karto v več barvah, je nujno poprej določiti barve za posamezne originale (potrebni so štirje). In še prednost te tehnike: podatke je mogoče pripavljati z vsakim računalnikom, če jih je ta zmožen pretvoriti v zapis, ki ga razume risalnik. Zdi se mi, da bi tudi mali spectrum zadoščal za takšno opravilo.«

Odpori zopar grafični tamatski prikaz

Računalniški obdelavi podatkov sledi fotografska. Sliko prenesejo s papirja na film in jo tako pripravijo za kopiranje na tiskarski plošči. Med risanjem risalnika in fotografsko preslikavo lahko nastanejo napake, zaradi hitrosti risalnika nemara niso narisane vse pike ali pa so nekatere sive. Delome odpravijo to napako s krepitvijo električnega naboja in z okrepljenim dovajanjem tonerja v risalniku, vender to naredi »senče« na vseh počrnjenih elementih slike. Kar skušajo pri fotografski obdelavi odpraviti te sence, se utegne primeriti, da na filmu izginejo vse pike, ki so na papirju sive, tako da barva po končanem tiskanju ne ustreza izbrani barvni na začetku. Takšne zadrege preženejo z dobro rektifikacijo risalnika in z uporabo svežega risalniškega papirja. Sicer pa napake odpravljajo že med fotografsko preslikavo na film, pri rastrih pa to dosežejo s pomanjševanjem originalov.

Edino tiskanje poteka na klasičen način.

Zapleteno? Kartografi, tisti, ki se sučejo okrog računalnikov, pravijo da ni. Razmišljajo o prihodnosti avtomatske kartografije, ki da je v »v rokah« še boljših risalnikov, zlasti laserskih in v satelitski tehniki, ki je »zelo pogodu rastrskemu risanju tematskih kart«. In podobno kot klasični kartografi pravijo, da kakovost kart – računalniških in klasičnih – ni odvisna samo od osnovnih podatkov, ki naj bi jih zbirale številne službe in organizacije, ampak zlasti od denarja. »Kolikor denarja nameni za izdelavo karte založnik, takšna bo karta. Nekaterih kart smo se zato že sami sramovali ...«

Če bi sodili po sežanskem računalniškem atlasu, potem kartografom in njihovim sodelavcem ne bo treba zardevati pri atlasih Slovenije in Jugoslavije, slednja utegneta prav zaradi izredne preglednosti, ki jo omogočajo grafične ponazoritve, hitreje prenesti razpravo o razvojnih razlikah s političnih vrhov v delegatsko bazo.

Po desetletnem pohodu je naša avtomatizirana kartografija pravzaprav šele oživela, toda še vedno mora premagovati »pasivni odpor«, kot pravijo kartografi, in nazadnještvo grafične tematske ponazoritve bi morali znati brati, tega pa ne učijo na nobeni osnovni šoli. Nič čudnega, če je zato ob predstavitvi sežanskega računalniškega atlasa eden od njegovih snovalcev sebi v brado zamrmral zadnje besede slovitega Galilea: »In vendar se vrti!« Računalniški atlas občine Sežana se nemara zdi majhen korak, toda tehnologija njegove izdelave je naredila tolikšnega, da je na tem področju ujela razvoj v industrijsko razvitih državah. Začelo se je vrteti.

»Z zaviranjem uvoza si družba žaga vejo . . .«

Dr. Stojan Zaler je eden tistih slovenskih strokovnjakov, ki jih je življenjska pot zanesla v svet v »zlati dobi« za fizike, elektrotehnike, matematike, astronome, se pravi v dobi, ko je Sovjetska zveza s sputnikom prehitela Američane in so ZDA v mrzlični tekmi za prestiž povsem sprostile finančne vire za razvoj znanosti. Dr. Zaler dela že 15 let v razvojnem oddelku IBM kot svetovalec za fiziko. Trenutno se ukvarja s problematiko mikroelektronskega pakiranja. Med njegovim bivanjem v domovini se je za reviji Moj mikro in Teleks pogovarjal z novinarko **Mojco Vizjak-Pevšič** in s sodelavcem glasila Iskra **Borisom Černom.**

Leta 1963, ko se je v ZDA že uveljavil kot priznan strokovnjak na področju elektronike, so dr. Zalerja povabili v Slovenijo, da bi vodil na novo ustanovljeni Laboratorij za mikroelektroniko pri Iskri. Z družino je preživel v Ljubljani eno leto, ker pa opreme ni in ni bilo, se je odločil za vrnitev v ZDA, kjer so ga sprejeli z odprtimi rokami. Kaj meni danes o naši elektronski industriji?

»Iskra je imela šanso pred dvajsetimi leti, ko so bili na primer tudi Francozi na začetku. Takšni prelomnici smo priče tudi danes. Sedaj se lahko zgodovina ponovi. Če bo Jugoslavija zamudila, če se ne bo spopadla z novo tehnologijo, bo postala kolonija. Imam občutek, da so politični tokovi speljali politično revolucijo, za tehnološki razvoj pa nimajo pravega posluha. S političnimi ukrepi se žal ne da skočiti v novo tehnologijo. Treba je nekaj narediti! V ZDA se ocenjuje po rezultatih dela, ne pa po govorjenju ...«

In še: »Ker danes jugoslovanska industrija ni zmožna zapolniti tržišča s svojimi računalniki, menim, da bi bilo najbolj smotrno, če bi za kakšna tri leta povsem sprostili uvoz. Tri leta bi morali računalnike na veliko uvažati, hkrati pa naj bi Iskra intenzivno razvijala doberi osebni računalnik! Težave z uvozom so pravi zločin proti razvoju družbe. Z zaviranjem uvoza računalnikov si družba žaga vejo, na kateri sedi ...«

Nadaljujmo pogovor o nekaterih novostih na področju mikroelektronike, novicah, ki včasih vzbujajo asociacije na znanstveno fantastiko.

— Mikroelektronika postaja vse pomembnejše proizvodne panoge tudi pri nas in zato je vprašanje, kam bo usmerjen njen razvoj v naslednjih letih, vse zanimivejše. Dr. Zaler, vi to področje dobro poznate ...

»K cenejši proizvodnji in razvoju tehnologij, ki omogočajo snovanje zahtevnejših sistemov, bo industrija še težila. Zato je tudi razvoj usmerjen v zmanjševanje geometrij in velikosti, torej miniaturizacijo, kar bo stroške proizvodnje glede na posamezno spominsko enoto ali računsko funkcijo bistveno znižalo. Sicer pa bo v tem času prevladovala tehnologija CMOS, silicijeva bipolarna tehnologija, tehnologija v galijevem arzenidu, verjetno tudi tehnologije na osnovi Josephsonovega efekta, razvoj bo tekel še na področju optične elektronike in magnetnih mehurčkov.«

— Za snovanje zmogljivejših sistemov je pogoj hitrost reševanja posameznih funkcij, kar lahko dosežemo na več načinov?

»Da, poleg že omenjenega zmanjševanja geometrij so še druge obetavne poti. Računalniki na osnovi superprevodnosti elementov v bližini absolutne ničle bi bili lahko v tem pogledu izredno zanimivi. Tu bodo preklopni časi v območju pikosekund (tisočink milijardinke sekunde).

Pri sobni temperaturi pa bo v območja pikosekund lahko posegla le tehnologija v galijevem arzenidu. Malo verjetno je, da bo take hitrosti mogoče doseči v siliciju. Bipolarna tehnologija pa prehaja v hitrosti ene milijardinke sekunde in bo prešla v območje pikosekund, vendar ne pod 200 ali 300 pikosekund.«

— Gibljivost elektronov je v galijevem arzenidu desetkrat večja kot v siliciju, kar v bistvu omogoča gradnjo desetkrat hitrejših računalnikov. Vendar je proizvodnja na osnovi galijevega arzenida dosti zahtevnejša in zato zelo draga ...

»Pri siliciju je ugodno, da lahko vlečemo iz njega monokristale premera 5 do 6 col (tj. do 15 cm), z galijevim arzenidom pa sedaj največ do 3 cole. So pa že kažejo rešitve tudi za ta problem. Z nanosom enovrstnega sloja galijevega arzenida na keramično ploščico ali na monokristal bi ga že lahko rešili, z epitaksijo molekularnih curkov.«

— Cenovno vzeto ima proizvodnje na osnovi silicija dokajšnjo prednost. Bo zato silicij ostal najpomembnejša surovina za mikroelektroniko?

»Silicij bo še zelo dolgo prevladoval. Le tam, kjer bodo preklopna hitrosti posameznih naprav odločilne, kot na primer v nekaterih vojaških projektih, bo verjetno prevladala tehnologija na osnovi galijevega arzenida. Ima pa galijev arzenid še nekatere lastnosti, ki jih bo moč s pridom izkoristiti. Nekatere diode iz galijevega arzenida lahko generirajo laserski žarek, ki je že sam osnova za prenos informacij. Tako bo tudi možno integrirati elektronsko in fotonsko tehnologijo, kjer pri eni prenašajo signal elektroni, pri drugi pa fotoni. Za Jugoslavijo bi bilo povsem dovolj, če bi uspešno osvojila tehnologijo na podlagi silicija.«

— Kdo se ukvarja z razvojem mikroelektronike na osnovi galijevega arzenida?

»V ZDA delajo na veliko, vendar ne dovolj intenzivno, na Japonskem pa intenzivno, vendar ne na veliko, tako da je to področje še precej neraziskano in lahko s tem v zvezi pričakujemo nekatere presenečenja.«

— In kakšen pomen ima germanij?

»Spet postaja zanimiv, ker omogoča delo v mikrovalovnih frekvencah. Torej omogoča prenos več podatkov na časovno enoto in s tem hitrejše delovanje sistemov.«

— S katero tehnologijo se bo začela peta generacija računalnikov in kakšne operativne hitrosti bodo zanjo potrebne?

»Po mojem mnenju bodo to izjemno zmogljivi računalniki, ki bodo zmožni simultanega procesiranja. To pomeni, da bi lahko istočasno delali po več programih in da bodo ti programi interaktivni, tako da se bo računalnik te vrste že približal paralelnosti, ki je značilna za človeške možgane. Doslej poznamo delovanje računalnikov le na osnovi programov sekvenčne narave v eni dimenziji, v peti generaciji računalnikov pa bodo programi potekali vzporedno in interaktivno.

Če na primer človek sliši pok od zadaj, se obrne in pogleda nazaj. Tu bi lahko rekli, da gre za slušno in vidno procesiranje, ki teče istočasno in se ob poku poveže tako, da človek obrne glavo. Seveda gre pri človeku za veliko bolj kompleksne procese.

Za peto generacijo računalnikov bo zato, poleg tehnologije najhitrejših tranzistorjev, nujna tudi taka programska oprema, ki bo omogočala paralelno procesiranje, pri čemer bo izredno pomembno še vprašanje interakcij. Ob slednjem pa se že pojavi pojem umetne inteligence.«

— Menite, da je možno dati računalniku tudi take inteligence, ki bi lahko bile intuitivne?

»Po mojem mnenju ne. Vsaj v času današnjih generacij še ne. Kaj bo v prihodnosti, pa je vprašanje, ki za sedaj presega znanstvene okvire in prehaja na področje znanstvene fantastike in poskusnih hipotez.«

— Kakšne hitrosti bodo računalniki pete generacije le morali doseči?

»Preklopni čas bo vsekakor moral biti pod eno nanosekundo (milijardinko sekunde). Zdaj pa imamo polprevodniške elemente, ki omogočajo preklopni čas v mejah med 2 in 20 nanosekundami. Nekateri gledajo na to tehnologijo

z vidika ločljivosti (izdelave) najmanjšega elementa gre tudi za vprašanje, v kateri mikronski tehnologiji bo to možno doseči. Odgovor je vsekakor v manj kot enomikronski.«

— Do kod bo možno zmanjševanje v siliciju?

»Tja do 0,1 mikrona, vendar te meje ne bo tako lahko doseči, ker se pojavijo fizikalne omejitve v litografiji. Izdelava je predvsem odvisna od kvalitete litografije. Iskrina litografije trenutno ne more delati v manj kot dvomikronski tehnologiji.«

— Kaj pa računalniki pete generacije na osnovi galijevega arzenida?

»Tam to ne bo tako kritično, ker je, kot rečeno, gibljivost elektronov desetkrat večja.«

— Predvidevate, da se bodo na področju mikroelektronske tehnologije pojavili tudi polimeri in organske substance?

»Da, vsekakor. Tudi sam sem delal dve leti na tem razvoju. Predvsem poznam tanke polimidne plasti, ki imajo visoko temperaturno obstojnost. Lahko jih kar eno uro obdržimo na 400 stopinjah Celzija. Toda pomembnejša je njihova obstojnost pri 100 stopinjah C. Če računamo, da bodo integrirana vezja delovala pri denimo 100 stopinjah C, moramo biti pozorni predvsem na ta vidik. Te snovi imajo tudi nizko dielektrično konstanto, torej je širjenje elektromagnetnih signalov v njih hitrejše.«

— Bo možno graditi računalnike na osnovi organskih snovi?

»Da. Menim, da bo sčasoma to možno. Področje je še v povojih, vendar znanstveniki že preučujejo posebne proteine, ki so sposobni preklapljenja. Gradnja bioračunalnikov je možna, čeprav je njihov čas za sedaj še precej odmaknjen, vendar pot je odprta. To bodo potem računalniki na osnovi ogljika in ne silicija.«

— Kaj pa računalniki na osnovi optične tehnologije?

»Tudi to je možno. Optični žarek iz monokromatskega laserja je možno modulirati in s tem prenašati podatke. Gradnja optičnega računalnika je zato možna, vendar bo to draga pustolovščina.«

— V zadnjem času je na področju računalništva naredila velik korak Japonska, medtem ko Evropa zaostaja ...

»Japonska je nevaren tekmec, ki ga Američani resno jemljejo. Sicer menim, da Američanov ne morejo prehiteti, bodo pa krepko vplivali na potek dogajanj na tem področju. Drugače je z državami EGS. Te še občutno zaostajajo. Imajo sicer nekatere nacionalne projekte, s katerimi želijo zmanjšati zaostajanje za Japonsko in ZDA. Predvsem želijo razvijati računalnike pete generacije. V tem smislu že ukrepajo vključno z investicijsko politiko. Tako zastavljeno delo bo že moralo dati nekatere rezultate, vendar bodo najverjetneje še naprej zaostajali za ZDA in Japonsko.«

— Kako pa so Japonci zakorakali v osvajanje mikroelektronske tehnologije?

»Japonci imajo v bistvu zelo usmerjeno gospodarsko politiko. Vsa njihova podjetja, ki so na videz samostojna, financirajo med sebojno povezane banke. Zato lahko vse velike državne projekte usmerjajo preko nacionalne banke, torej iz enega centra, kjer je združen bančni, podjetniški in državni kapital. Tako so se usmerili v projekte, kot je razvoj najboljše industrije fotoaparatov, radioaparatov in televizorjev ter ta projekt uspešno uresničili. Podobno so se lotili tudi proizvodnje manjših avtomobilov in tudi s tem projektom uspeli. Sedaj pa vse svoje potenciale usmerjajo v gradnjo računalniške industrije in seveda s tem mikroelektronike, ki je osnova za izdelavo računalnikov. Očitno so se v ta zadnji, lahko rečemo nacionalni projekt, zagnali še z večjo vnemo kot v prejšnje.«

— Torej so na področju računalništva in mikroelektronike zdaj v igri vlaganja na nacionalni ravni?

»Da, država, ki ne vlaga v ta področja, je ,neumna' država. Vidim, da v Iskri in drugih delovnih organizacijah v Jugoslaviji tudi osvajajo to področje. Vendar je vse to še bistveno premalo, ker ni dovolj vlaganj v projekte. Vsaj tri odstotke družbenega proizvoda bi moralo iti za pospeševanje sodobne elektronske industrije.«

— In kaj bi torej rekli tistim našim možem, ki lahko kakorkoli vplivajo na usmerjanje naših investicij?

»Naj namenijo pet odstotkov družbenega proizvoda za pospeševanje elektronske industrije. To se bo bolj obrestovalo, kot si morda kdo misli.«

Zboljšujemo spectrum

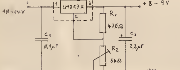
IZTOK SAJE

Sinclair ZX spectrum je odlična naprava, poceni in zelo zmogljiva. Poglavitno lastnost, nizko ceno so dosegli tako da je v računalnik vdelano samo najnujnejše, vse polno je kompromisov in poceni rešitev.

Z malo dobre volje, časa in znanja lahko spectrumu dodamo marsikaj, kar so načrtovalci izpustili ali česar sploh niso predvideli. Preden opišem nekaj takšnih malenkosti, po katerih bo postal vaš računalnik veliko bolj prijazen, nekaj besed o tem, kako je narejen.

Hišni računalniki so zelo občutljive naprave. Integrirana vezja pregorijo zaradi previsoke napetosti na priključkih ali pregrevanja nad dovoljeno temperaturo, povezave med elementi se rade prekinejo, tipkovnica in ohišje se hitro poškodujeta, če smo pregrobi.

Kadar računalnik odpremo, moramo dobro vedeti, kaj želimo spremeniti in kako bomo to naredili. Če ne znate spajkati, je bolje, da to prepustite izkušenemu prijatelju. Drobne kontakte in povezave hitro pregrejemo in uničimo, zato moramo uporabiti šibak spajkalnik z močjo do 25 vatov. Novi elementi morajo biti preverjeni in zanesljivi, drugače nam naredijo več škode kot koristi. Prav tako moramo trikrat preveriti vse povezave, da ne poškodujemo elementov ali računalnika. Večino dodatkov priključimo na računalnik prek za to namenjenega spojnika (konektorja). Posebno moramo paziti, da nikoli ne vstavljamo spojnika v vključen računalnik. Če ima dodatek lastno napajanje, moramo poskrbeti, da bo računalnik vedno vključen pred dodatkom ali hkrati z njim.



Skica 1 stabilizator napetosti

Program za 50-baudni teleprinter!

Naslovi Vrednosti!

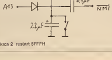
23296	3	3	25	14	9	1	13	24	20	6
23306	11	15	10	20	12	24	22	23	10	5
23316	16	7	30	19	29	21	17	47	61	50
23326	57	33	4	60	37	62	57	61	49	37
23336	47	50	29	49	44	33	60	61	84	55
23346	51	33	42	48	53	39	38	56	46	46
23356	47	62	58	87	32	17	1	231	118	211
23366	127	197	6	6	118	31	48	4	211	223
23376	24	2	211	127	118	16	244	211	223	193
23386	110	205	84	21	216	207	12	43	71	230
23396	32	40	24	166	40	13	126	230	32	117
23406	120	35	203	67	91	53	40	74	201	62
23416	27	205	67	91	112	24	235	62	32	166
23426	40	230	62	31	24	241	205	204	91	126
23436	230	15	32	248	201	35	52	43	33	82
23446	201	33	66	91	53	192	52	43	254	13
23456	86	238	40	28	254	22	56	238	284	32
23466	40	32	56	227	254	120	40	30	254	96
23476	56	2	230	223	230	63	229	111	126	225
23486	24	141	84	64	62	0	205	67	91	62
23496	2	195	67	91	62	4	24	160	214	165
23506	56	240	17	149	0	245	205	65	12	212
23516	204	91	26	230	127	205	151	91	26	19
23526	133	40	245	209	254	72	40	3	254	130
23536	214	122	254	3	216	24	213	33	151	91
23546	34	197	02	24	190					

Vsota vseh zlogov! 23653

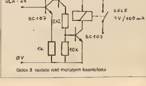
Program za 50-baudni teleprinter

Pregrevenje lehko preprečimo

Prva večja pomanjkljivost spectruma je pregrevanje. Po nekaj urah delovanja je v ohišju tako vroče, da se čudimo, kako da se vse skupaj ne vžge. To je tudi najmočnejši vzrok pogostih okvar. Računalnik se pregreva zaradi prevelike napetosti napajalnika. Ve-



Skica 2 restart 5FFFH

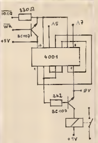


Skica 3 nadzor nad motorjem kasetofona

čina napajalnikov ima namesto predpisanih 9 V precej več, tudi 14 voltov sem že izmeril. To pomeni, da v računalniku pretvarja v toploto nekaj vatov več, kot je treba. Zato moramo v napajalnik vdelati stabilizator s skice 1 ali kaj podobnega.
Integrirano vezje LM 317 je stabilizator napetosti. Izhodno napetost mu nastavljamo s spremenljivim uporom R 2, upor R 1 je za primerjalno napetost, kondenzatorja pa zagotovita zanesljivejše delovanje. Izhodno napetost nastavimo med 6 in 9 V. Če je premajhna, računalnik ne deluje ali pa zagledamo temno progo, ki se počasi giblje čez zaslon. Integrirano vezje LM 317 moramo vgraditi na dovolj veliko hladilno rebro, ki ga dodamo na zadnji strani napajalnika. Zadostuje tudi kos pločevine, večji od 40 kvadratnih centimetrov. Za druge elemente je dovolj prostora v samem napajalniku. Dodamo lahko stikalo za vklop računalnika in svetlečo diodo (LED), ki nas opozarja, da je računalnik vklopljen. Kmalu boste videli, kako upravičena je ta sprememba. Hladilnik se segreva in vsa ta toplota bi bila drugače v računalniku.

Tipka za rastart

Večina mojih znancev je dodala spectrumu tipko za restart, tako da ni treba izklapljati napajanja, kadar želimo izbrisati pomnilnik. Slaba plat takega restarta je, da izgubimo vsebino pomnilnika. Toda tudi proti temu je zdravilo. Skica 2 kaže shemo vezja za restart 5FFFH. S pritiskom na tipko sprožimo NMI prekinitev mikroprocesorja in ga prisilimo, da začne izvajati program, ki je na naslovu 0066H. Ker je v sistemskih spremenljivkah 23728 in 23729 vedno O, procesor skoči na naslov O (podobno kot z restartom, le da pri tem ni izključeno osveževanje dinamičnega pomnilnika). Pred diod D 1 in D 2 vežemo naslovni vodili A 13 in A 15 na maso. Tako izklopimo zgornjih 40 K pomnilnika in njegova vsebina se ne spremeni. Po restartu lahko popravimo, posnamemo ali znova poženemo program v zgornjih 40 K. Tipko moramo držati toliko časa, da je restart končan in se nam na zaslonu pokaže nenavadna slika, ne pa znani napis 1982 Sinclair Research. Potem popravimo sistemski spremenljivki UDG (23676) in P-RAMT (23733) ter nadaljujemo delo. Enak učinek dosežemo, če namesto NMI uporabimo nožico RESET v mikroprocesorju in zagotovimo, da bo impulz za restart sinhron s signalom M 1. Tekšno vezje je opisano v večini knjig o mikroprocesorju Z 80. Dodati moramo dve integrirani vezji, pa lahko ustavimo računalnik tudi med programi, ki uporabljajo sistemsko spremenljivko na naslovu 23728 (npr. Beta Basic).



Skica 4: enokanalni vmesnik

Povezava s kasetofonom

S spectrumom lahko nadzorujete kasetofon. Če imate kasetofon s priključkom za izklop motorja (El reporter ipd.), ga povežite z relejem vezja s skice 3. Vezje je povezano z nožico 26 integriranega vezja ULA v spectrumu. Ko vklopimo računalnik, rele izklopi motor za toliko časa, dokler ne pritisnamo katerekoli tipke. Prav tako izklopi motor kasetofona, ko naložimo vse dele kakega programa. Tako nam ni treba čakati ob računalniku, kdaj naj izklopimo kasetofon. Če se program začne z ukazom BEEP, pa vezje ne deluje, saj je nožica 28 ULA vezana na spectrumov brenčač. Z malo eksperimentiranja se boste navadili uporabljati ta pripomoček.

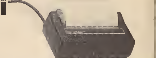
Pogosto si želimo povezati računalnik z drugimi električnimi napravami. Najpreprostejši enokanalni vmesnik vidimo na skici 4. Sestavljajo ga enobitna pomnilne celice in vezja za spreminjanje stanja. Z ukazom OUT 223 N vklopimo, z ukazom OUT 127 N pa izklopimo rele. Ker vezje ni vezano na podatkovno vodilo, je N katerokoli število med O in 256. Z opisanim vezjem lahko preklapljamo dva kasetofona, kadar presnemavamo kasete. Računalnik lahko ob določeni uri vklopi in pozneje izklopi radio ter sam posname program Radia Študent, prav tako lahko izbira telefonska številke. Skratka, tudi tu velja tisto kot pri programiranju, računalnik je omejen samo z vašimi idejami in znanjem.

Tiskalnik iz starega teleprinterja

Če lahko kje dobite star teleprinter s hitrostjo 50 bd, ki je že odbrenkal svoje v teleprinterskem omrežju, ga lahko priključite na spectrum po opisanem vmesniku. Ker rele ni dovolj hiter, morate namesto BC 107 uporabiti močnejši tranzistor NPN (2 N 1893 ali podoben) in z njim prekinjati linijski tok iz teleprinterja ali napajalnika. Program za teleprinter vtipkajte v računalnik in poženite z RANDOMIZE USR 23543. Imeli boste tiskalnik, ki bo ubogal ukaza LPRINT in LLIST, ukaza COPY pa ne. Prav tako ne razume ukazov TAB in AT. Vrstica je dolga 64 znakov. Programa vam ne bo težavno disasemblirati, če vas zanima, kako deluje. Na začetku je tabela za pretvarjanje ASCII v kodo CCITT-2, sledijo pa program za tiskanje, program za pretvarjanje kod, program za obdelavo gesel in nadzornih kod ter program za inicializacijo kanala »p«. Naredite, preizkusite in videli boste, kako enostavno in poceni ste prišli do uporabnega tiskalnika. Program je dolg 255 zlogov (bytov), začetni naslov je 23296.

Še enkrat pa vam svetujem, da ne predelujete računalnika, če se ne spoznate na elektrotehniko. Priskrbite si material in poiščite kakšnega strokovnjaka, da vam bo naredil kar želite.

Kako očistiti tiskalnik ZX



MIRKO TISNIKAR

Tiskalnik ZX (ZX printer) je prva v vrsti pogruntavščin, pri katerih se Sinclair z vsemi štirimi otepa gibljivih delov. Dejansko so ti pri mehanskih ne prevah najpogostejši vir okvar.

Napravica je morda smešno neugledna, toda za zares majhen denar (v ZRN dobite ta tiskalnik ža za manj kot 150 DM) boste lahko rezultate programov za vedno shranili. Tudi popravljanje na papirju je čisto nekaj drugega kot iskanje napačne vrstice na zaslonu.

Tiskalnik dela precej podobno kot se prenaša slika iz pomnilnika na zaslon. Če imate ostro TV sliko, boste opazili, da je zaslon sestavljen iz množice horizontalnih črt. Nekateri deli so črni, torej je tam nekaj narisano, nekateri pa beli, prazni. Tudi izpis na tiskalniku ZX se sestavlja iz horizontalnih črt.

Motor premika jermenček, na katerega sta pritrjeni dve igli. Igli skoz rezo izmenoma drgneta po metaliziranem papirju. Če spustimo skoz papir in iglo električni tok, preskoči iskra in osmodi del papirja. Ta koščok je videti črn v primerjavi z deli, ki niso zažgani. Po prehodu prve igle se papir neznatno premakne naprej in že je tu druga igla, ki bo izžgala naslednjo linijo. Osem takih linij sestavlja en znak, 176 pa sliko zaslona.

Računalniki delajo ali pa ne. Tiskalnik lahko dela bolje ali slabše. Z malo spretnosti lahko kvaliteto tiskanja precej zboljšamo.

1. Izpis je neenakomeren, na nekaterih mestih zelo bled

Največkrat povzroči to neenakomerna debelina kovinske plasti na papirju.

2. Tiskalnika dlje časa niamo uporabljali, zdaj pa ne potiane papirja skoz režo

Papir se je na valju deformiral. Odprimo tiskalnik, kot da želimo vložiti nov zvitek, odvijmo nekaj centimetrov z zvitka in namestimo zvitek nazaj.

3. Stolpci v različnih vrsticah niso drug pod drugim, znaki so zato »rezsuli« in težko čitljivi

Potrebujemo križni izvijač in šolski čopič. Ker ostaja v tiskalniku mnogo sežgenih ostankov papirja (saj), pogrnite mizo s časopisom ali papirnato brisačo.

Najprej bomo očistili optični disk. Ta namreč z lego kovinskih žarkov pomaga določiti natančno lego pisalne konice. Če so žarki umazani, ne odbijajo več svetlobe in izpis postane nečitljiv. Odvijemo torej vijake »a«. (Pazljivo, da ne pretrgemo ali izpulimo kakšne žice.) Nato s čopičem odstranimo vse saje in koščke metaliziranega papirja. Še posebej pazite na optični disk. Če ga boste poškodovali, sami ne boste več mogli popraviti tiskalnika.

4. Podmezovanje gibljivih delov ne mažite z oljem ali s čim podobnim!

Na mestni površini se dosti laže nabira nesnaga. Še huje pa bo, če boste zamestili optično kolesce. V skrajnem primeru lahko s koščkom masla podmažete obe osi kolesca, na kateri je napeta »gosenica« z iglami. Zobata kolesca in druge prenose lahko posujete z nekaj malega gratitnega prahu. Ošiite ne preveč trd svinčnik (HB do 2 B) nad kolesci, potem pa s čopičem razporedite prah in odstranite odvečnega.

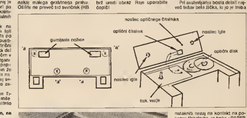
Če ste imeli do tod kaj težav z debelimi in nerodnimi prsti, raje tiskalnik spet zaprite. Še posebej pazite, da je rdeča žička nepoškodovana in optični del čist. Ko je tiskalnik sestavljen, preizkusite, ali deluje pravilno.

Če pa se upate nadaljevati, odstranite nosilec papirja, kot je to zapisano v priročniku za tiskalnik. Zataknilo se bo zaradi bele žičke. To je treba pazljivo sneti iz kontakta na zgornji strani tiskalnika. Opazili boste, da je v tem delu še dosti več umazanije kot spodaj. Če jo boste poskušali odstraniti z močnim pihanjem, si boste morali brž umiti obraz. Raje uporabite čopič!

Med čiščenjem tega dela še posebej pazite, da ne boste poškodovali nosilcev pisalnih iglic. Te so najobčutljivejši del tiskalnika. Če piše tiskalnik medlo, včasih pomaga, če iglici razmaknamo, tako da pritiskata z večjo silo na papir.

Na kakovost izpisa močno vpliva tudi čistoča reze, skoz katero iglici pišeta. Umazanija se reže še posebej trdovratno drži in s čopičem ne bomo ničesar opravili. Pomagamo si lahko z nohtom, v skrajnem primeru pa saje nerahlo odpikmo z izvijačem. Tudi ploskev, po kateri drgne druga konica pisalne glave, mora biti zaradi čim boljšega kontakta karseda čista.

Pri sestavljanju bosta delali največ težav bela žička, ki jo je treba



natakniti nazaj na kontakt na pokrovu tiskalnika, in tipka »PAPER FEED«, ki jo moremo pazljivo vtakniti v odprtino.

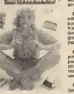
Ne koncu očistimo še nosilec papirja. Gumijasti valj, ki vleče papir, lahko poplaknemo z nekaj kapljicami alkohola.

Če tiskalnik kljub navedenim popravilom še vedno ne dela, kot bi moral, bo treba servis prepustiti električarju. Izpis je najpogosteje bled zaradi preslabega napajanja v računalniku (skrajne posledice so zvijanje slike med tiskanjem in spremembe barve roba) ali pa ima tiskalnik preveliko porabo. Prav zato, ker je poganjanje umazanih kolesc požrešnejše, svetujemo servis po vsakem porabljenem paketu papirja (5–6 zvitkov).

Iz vsebine dosedanjih številk revije Moj mikro

V številkah od junija pa doslej smo med drugimi objavili naslednja članka o računalnikih njihovih dodatkih in programe

SINCLAIR – 16 K, 48 K, ZX 81 in QL razširitev spectruma na



80 K (november na straneh 22 in 23), razširitev spectrumovega spomina (september na strani 27) tipka za čiščenje spectruma (november 1984 na strani 23) igralne palice za spectrum (september na strani 28), video izhod spectruma (junij na strani 45), tabela programov za 16 K, 48 K in ZX-81 (september na straneh od 29 do 40 in oktober na strani 26) razširitev basica v spectrumu (september na straneh od 41 do 43) uporabni program Beta Basic (oktober na strani 42), računalnik kot telefon (oktober na strani 22) spectrum v statiki gradbenih konstrukcij (julij–avgust na strani 16), primerjava spectruma in QL (julij–avgust na strani 21 in 22), sposojeni test za sinclair QL (junij na straneh 18 in 19) in supertest računalnika QL (november na stranah 4 5 in od 14 do 16), programi za Sinclairove računalnike so objavljeni v vseh dosadanjih številkah mesečnika Moj mikro

COMMODORE priključitev commodorja na običajni kasetofon (junij na strani 44) praktnitve na CBM-64 (september na straneh 38 in 39): uporabni program Simon's Basic (oktober na straneh 43 in 44) uporabni programi za commodore – disassembler za 6510 (6502) grafični programi, o pretvorniku števil itd (november na strani 29 oktober na straneh 30 do 32, julij–avgust na straneh 37 in 38) junij na straneh od 36 do 40)

TEXAS INSTRUMENTS PC ekskluzivni test (september na straneh 4 in 5)

SHARP ekskluzivni test računalnika sharp MZ 731 (november na straneh 6 in 7) sposojeni test računalnika sharp PC-5000 (september na strani 10) programi za Sharpove računalnike (november na straneh 31 in 32)

HEWLETT PACKARD sposojeni test računalnika hewlett packard 150 (julij-avgust na straneh od 4 do 7) sposojeni test računalnika HP-110 (november na strani 11)

MACINTOSH ekskluzivni test (junij na strani 8)

BBC predstavitev tega računalnika (junij na straneh 50 in 51) ekskluzivna predstavitev računalnika

ACORN ELECTRON (november na strani 10)

MEMOTECH: sposojeni test računalnika memotech MTX 512 (julij–avgust na straneh 8 in 9)

IBM PC: ekskluzivni test in predstavitev (oktober na stra-

nah 8 in 9 ter november na straneh od 46 do 51)

ORIC ATMOS ekskluzivni test (oktober na straneh 8 in 9)

AT: predstavitev (november na straneh 12 in 13)

ISKRA predstavitev domačega računalnika HR 84 (junij na strani 9)

GALAKSIJA test (julij–avgusta na straneh 14 in 15) in širitev tega računalnika (september na strani 21)

COLOR GRAF predstavitev (junij na strani 8)

Objavili smo tudi primerjalno tabelo najbolj razširjenih mikroračunalnikov in tiskalnikov

spodnjega razreda (junij na straneh 34 in 35) in sposojeni test hitrosti in natančnosti različnih računalnikov (oktober na strani 35). podrobnosti o številnih drugih računalnikih pa smo objavili v reportažnih zapisih s sejmov v Londonu in na Dunaju (oktobrska in novembrska številka)

Med hardwarskimi nasveti in dodatki, objavljenimi v reviji Moj mikro naj opozorimo na naslednje

– splošno o **tiskalnikih,** tiskalnika seikosha GP 100 VC in gemini (september na straneh od 6 do 9)

– **datoteka,** splošno o mikrotračnih enotah, microdrive in interface 1 (junij na strani 24 in julij–avgust na straneh 42 in 43) disketna enota VIC-1541 (julij–avust na strani 47)

– **tipkovnice.** testi Lo Profile Fuller FDS, Transform Dk Tronics, Premk, Špica Ines (november na straneh od 52 do 54)

– prekrivala za tipkovnice (november na strani 42)

– pomnilniški modul za VIC-20 (september na strani 24)

– vmesni RS 232 C (september na straneh 25 in 26)

– računalnik kot telefon (oktober na strani 22).

– splošno o dodatkih in njihovi izbiri (oktober na straneh od 52 do 54)

NAROČAM *ravijo* **MOJ MIKRO**

Naročnino bom plačal
po prejamu položnice

(ime in priimek) (ulica, hišna številka)

(poštna številka) (pošta)

(podpis)

Zvaza organizacij za tehnično kulturo Slovenija
bo v prvi polovici dacembra latos predstavila dve novi kasati za ***MAVRICO 48 K:***

»Cicibanova abaceda« *avtorja Davorja Bonačića, ki je namanjena mlajšamu bratcu ali sestrici, da bi spoznal(a) črke in se naučil(a) sestavljati besede. Mimograda bo spoznal(a) računalnik in mu (ji) na bo več navaren (navarna).*

»Anglaško-slovenski slovarček« *avtorjev Primoža Jakopina, Janaza Kaniča in Borisa Krambergerja. Naslov pove skoraj vse: besed pa je kar 3175. Kaseti je priložena knjižica.*

Cena prve kasete je 800 din, druge s knjižico vred pa 900 din. *Kupite jih lahko v vseh knjigarnah, zlasti pri Mladinski knjigi.*

Dobite lahko tudi ponatis prve kasete ***Kontrabant I,*** *na kateri pa je nekaj programov v srbohrvaščini. Cena je 1300 din.*

SE ŽELITE NAROČITI?

Ker je bil **MOJ MIKRO** v kioskih že večkrat razprodan, ves vabimo, da se naročite nanj. Izpolnjeno naročilnico pošljite na naslov: **ČGP Delo, oddelek za naročnine, Titova 35, 61000 Ljubljana.**

Po prejemu naročilnice vam bomo poslali položnico (naročninska cena 1200 din za pol leta, 2400 din za vse leto) in nato boste Moj mikro redno dobivali na svoj naslov.

SPOROČILO OGLAŠEVALCEM

MOJ MIKRO bo začel po novem letu izhajati tudi v srbskohrvatskem jeziku. Vsak oglas bo prebralo daleč več bralcev. Zato smo cene malih oglasov, ki so bile doslej reklamno nizke, izenačili s cenami oglasov v drugih izdajah TOZD Revije in v dnevniku Delo.

Cena malega oglasa do 10 besed je 300 din. Vsaka nadaljnja beseda stane 25 din.

Vse informacije o oglasih z daljšo vsebino in večjo, poudarjeno obliko posreduje uredništvo Teleksa po telefonu (061) 319-280.

Revija MOJ MIKRO *skupaj z zastopniki in proizvajalci računalnikov razpisuje:*

RAZPIS

za izvirne programe jugoslovanskih avtorjev, namenjene računalnikom **ZX spectrum, commodore 64** in **sharp MZ 700.**

Vsebina in namen programov nista omejana. Programe bomo ocenjevali v treh skupinah za vsak tip računalnika:
- **izobraževalni program**
- **uporabni programi**
- **igre**

Med programe za vsak računalnik bomo razdelili po tri nagrade:

1. nagrada 15.000 dinarjev
2. nagrada 10.000 dinarjev
3. nagrada 5000 dinarjev

Vse programe, ki bodo ustrezali kvalitetnemu nivoju za objavo, bomo ob soglasju avtorjev odkupili in izdali v najprimernejši obliki za distribucijo (kaseta, gibki disk).

Programi, poslani na natečaj, morajo ustrezati naslednjim pogojem:

1. na naslov Moj mikro, Titova 35, 61000 Ljubljana, s pripisom Za natečaj, morajo prispeti najkasneje do 1. marca 1985,
2. programi ne smejo biti še nikjer objavljeni ali kako drugače publicirani,
3. programi morajo biti v obliki, primerni za pregled,
4. skupaj s programom je treba poslati: navodila za uporabo, kratek opis programa (ideja), spisek uporabljanih pripomočkov in programov, ki niso v osnovnem bralnem pomnilniku računalnika (prevajalniki, drugi jeziki, rutine obstoječih programov)

Komisija v sestavi članov uredništva redakcije revije Moj mikro, priznanih jugoslovanskih strokovnjakov na področju računalništva in predstavnikov sponzorjev bo prejete izdelke pregledala najkasneje v 14 dneh po izteku razpisa. Odločitev komisije bo dokončna. Rezultati natečaja bodo objavljeni v obeh aprilskih izdajah računalniške revije Moj mikro.

Veselo na delo.
Čakajo vas nagrade.
Pot vašim izdelkom na police knjigarn je odprta.

PROGRAMI

Tudi v tej številki objavljamo nekaj zanimivih izpisov, ki so jih poslali naši bralci. Vse objavljene programe seveda honoriramo, med 1000 in 10000 dinarji, odvisno od dolžine in kvalitete.

Programe dobimo najraje na kasetah. Tudi listingi, ki jih je moč neposredno prefotografirati, so dobrodošli. Tiste pa, ki niso v taki obliki, moramo pretipkati, zato se lahko njihova objava nekoliko zavleče.

In ne pozabite na primerno spremno besedilo.

****

Kaset in izpisov ne vračamo po pošti, lahko pa jih dvignete v uredništvu.

Ker izpisujemo na matričnem tiskalniku, je izpis nekoliko drugačen, kot bi bil na ZX tiskalniku ali na ekranu. Širok je 48 znakov. Inverzni znaki so zapisani **mastno in so podčrtani**, UDG pa so natisnjeni *poševno*.

Upamo, da smo na ta način še povečali čitljivost in preglednost izpisov.

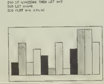
DIAGRAMI

Razmerja statistična stanja in razna gibanja najbolj nazorno predstavimo z diagrami. Naslednji program omogoča hitro upodabljanje spremenljivih veličin na dva načina: z empiričnimi krivuljami ali stolpci. Na začetku programa se odločimo za krivuljni/stolpični diagram, nato pa vstavimo druge podatke za formiranje diagrama. Največje število spremenljivk je v obeh primerih 6, največje število prikazov vrednosti vsake spremenljivke je pri krivuljnem diagramu 12, kar ustreza npr. upodobitvi mesecev v enem letu, pri stolpičnem diagramu pa je odvisno od predhodnega števila spremenljivk (od 6 do 41). Pri stolpičnem diagramu se odnos max/min avtomatičen in dobimo prikazano razmerje med njimi, ko moramo pri krivuljnem diagramu nastaviti območje vrednosti in lahko vrednosti nato tudi odčitavamo.

Diagrama sta dovolj nazorna v črno – beli tehniki, lahko pa ju opremite tudi z barvami. Ob naznačenem vstavljanju podatkov nam program sporoči, predno se izvede.

Miloš Korenč
Koper

```
10 BORDER 0: INK 7: PAPER 0
20 CLS
30 DRAW 0,175: DRAW 255,0: DRAW 0,-175: DRAW -
   255,0
35 BEEP .1,14: BEEP .1,17
40 INPUT "VRSTA DIAGRAMA kr/st";a$
50 IF a$="kr" OR a$="st" THEN GO TO 70
60 GO TO 35
70 IF a$="st" THEN GO TO 400

80 REM

81 REM krivuljni diagram

82 REM
90 INPUT "Obmocje vred 1<=50110?1";nbx
95 FOR t=0 TO 167 STEP ABS (169/nbx)
100 PLOT 0,t: DRAW 3,0
105 PLOT 255,t: DRAW -3,0
110 NEXT t
120 PLOT 3,169: DRAW 0,6
125 PLOT 252,169: DRAW 0,6
130 INPUT "Koliko spremenljivk? (1-6)";a
140 IF a<1 OR a>6 THEN GO SUB 360
150 INPUT "Koliko vrednosti? (2-12)";b
160 IF b<2 OR b>12 THEN GO SUB 360
170 FOR w=0 TO 255 STEP 255/(b-1)
180 PLOT w,0: DRAW 0,3
190 NEXT w
200 LET n=2
210 LET k=251/(b-1)
220 DIM m(a,b)
230 FOR x=1 TO a: FOR y=1 TO b
240 INPUT "spremenljivka ";(x);" vrednost ";(y)
   ;" = ";m(x,y)
245 IF m(x,y)<0 THEN PRINT AT 11,3; FLASH 1;"
   SAMO POZITIVNE VREDNOSTI": PAUSE 50: GO
   TO 240
246 IF m(x,y)>nbx THEN PRINT AT 11,7; FLASH 1;"
   PREVELIKA VREDNOST ": PAUSE 50: GO TO 240
250 LET m(x,y)=ABS (169*m(x,y)/nbx)
260 PRINT AT 11,3;"
270 NEXT y: NEXT x
280 FOR p=1 TO a: FOR q=1 TO b
290 IF n>=253+k THEN LET n=2
300 LET n=n+k
310 PLOT n-k,m(p,q)
320 IF q<b THEN DRAW k,m(p,q+1)-m(p,q): PLOT n-
   k,m(p,q)+p-1: DRAW k,m(p,q+1)-m(p,q)
330 NEXT q: NEXT p
340 GO TO 655
360 PRINT AT 11,12; FLASH 1;"NAPAKA!": BEEP 1,
   9: GO TO 20

400 REM

401 REM stolpicni diagram

402 REM

410 INPUT "Koliko spremenljivk?(1-6) ";a
430 IF a<1 OR a>6 THEN GO TO 360
440 INPUT "Koliko vrednosti?(1 - ";INT (246/a/6)
   ;") ";v
450 IF v<1 OR v>246/a/6 THEN GO TO 360
460 LET d=INT (246/a/v)-1
470 DIM o(v,a)
480 LET s=INT ((254-(a*v)*(d+1))/2)
490 LET max=0: LET min=0
500 FOR i=1 TO a: FOR c=1 TO v
510 INPUT "spremenljivka ";(i);".vrednost ";(c)
   ;"=";o(c,i)
515 IF o(c,i)<0 THEN PRINT AT 11,3; FLASH 1;"
   SAMO POZITIVNE VREDNOSTI": PAUSE 50: GO
   TO 510
520 LET max=(max+o(c,i)+ABS (max-o(c,i)))/2
530 LET min=(min+o(c,i)-ABS (min-o(c,i)))/2
535 PRINT AT 11,3;"
540 NEXT c: NEXT i
550 IF min=max AND min=0 THEN LET max=146
560 IF min=max AND min<>0 THEN LET min=0
570 FOR i=1 TO v: FOR c=1 TO a
580 LET o(i,c)=INT ((o(i,c)-min)*146/(max-min))
590 FOR j=s TO s+d STEP c
600 FOR u=20 TO 20+o(i,c) STEP c
610 PLOT j,u
620 NEXT u
630 NEXT j
635 PLOT s,20: DRAW 0,o(i,c): DRAW d+1,0: DRAW
   0,-o(i,c): DRAW -d-1,0
640 LET s=s+d+1
650 NEXT c: NEXT i
655 BEEP .1,17: BEEP .1,14
660 STOP
```



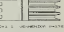
JERMENSKI PRENOSI

Program izračuna jermenski prenos s klinastimi jermeni.
Izračunavamo lahko moč do 150kW, hitrost jermena 25m/s in največji premer jermenice 1000mm, z 8 jermeni na jermenici.

Primer naloge:
Želimo izračunati jermenski prenos za ventilator: moč 5kW, motor 1450vrt/min, ventilator 2000vrt/min, medosna razdalja 750mm.
V program vstavimo podatke o pogonu iz naloge in dobimo izračun vseh važnejših podatkov o pogonu. Če nismo ustreznega jermena, lahko povečamo premer jermenic ali medosno razdaljo. Nato dobimo načrt za izdelavo jermenice na motorju in na ventilatorju.
Program je namenjen za spectrum 16 in 48 K.

Rado Jelnikar
Kresnice

```
2 REM Jermenski prenos
5 CLS : BORDER 2: PAPER 6: INK 0
7 PRINT AT 10,2; IZRACUN JERMENEGA PRENOSA
  PRINT AT 15,2; Programiral ": PRINT AT 17,
  10;"JELNIKAR RADO": PAUSE 300: CLS
10 PLOT 0,100: DRAW 250,0: PLOT 60,165: DRAW
  0,-135: DRAW 5,1: DRAW 0,-2: DRAW 5,1:
  DRAW 120,0: DRAW -5,1: DRAW 0,-2: DRAW 5,1:
  DRAW 0,135
11 CIRCLE 60,100,30: CIRCLE 160,100,60: PLOT
  60,130: DRAW 120,30: PLOT 60,70: DRAW 120,-
  30
12 PRINT AT 19,14; A": PRINT AT 18,5;"n1":
  PRINT AT 18,24 "n2 ": BEEP 1,3
20 INPUT "Moc prenosa v KW=";  N
25 INPUT "Stevilo vrt/min n1=";  na
30 INPUT "Stevilo vrt/min n2="   nb
35 INPUT "Zeljena medosna razdalja A=",  Y
38 BORDER 0: PAPER 7: INK 0
40 IF N<1 THEN LET D=50: LET a=8: LET Hj=5:
  GO TO 90
45 IF N>1 THEN GO TO 50
50 IF N<5 THEN LET D=90: LET a=13: LET Hj=9:
  GO TO 90
55 IF N>5 THEN GO TO 60
60 IF N<20 THEN LET D=125: LET a=17: LET
  Hj=11: GO TO 90
65 IF N>20 THEN GO TO 70
70 IF N<75 THEN LET a=25: LET D=240: LET
  Hj=16: GO TO 90
75 IF N>75 THEN GO TO 80
80 IF N<150 THEN LET D=355: LET a=32: LET
  Hj=20: GO TO 90
85 IF N>150 THEN LET D=500: LET a=40: LET
  Hj=25: GO TO 90
90 IF na<nb THEN GO TO 100
95 IF nb<na THEN GO TO 105
100 LET Db=D: GO TO 130
105 LET Da=D: GO TO 135
110 LET Da=Da+5: GO TO 135
115 LET Da=Da-5
117 LET D=D-((D/100)*70)
118 IF Da<D THEN PRINT AT 10,3; JERMENICA
  PREMAJHNA!": PRINT AT 15,5;"VSTAVI NOVE
  PODATKE ": PAUSE 200: GO TO 1
119 GO TO 135
120 LET Y=Y+5: GO TO 230
130 LET Da=Db*(nb/na)
131 LET Da=INT Da: GO TO 137
134 LET Db=INT Db
135 LET Db=Da*(na/nb)
136 GO TO 138
137 LET v=(((Da*PI)/1000)*na)/60
138 LET v=(((Db*PI)/1000)*nb)/60
140 LET v=INT v
144 RESTORE
145 DATA 2,0 6,0 8,1-5,2-8,3 3
146 DATA 3 0 [illegible] 9 3 4 4 4
147 DATA 4,0 4,0 8 1 2 1 4,4 5 6
148 DATA 5 1 1 4 [illegible] 4 8 6 8
149 DATA [illegible] 1 1 1 6 [illegible] 1 5 8 7 9
150 DATA 7 0 6 1 2 2 4 6 8 9
151 DATA 8 1 4 2 2 4 2 7 5,10
152 DATA 9,0 8,1 5 2 5 4 8 8 3 11
153 DATA 10 0 9 1 7 2 6 5 3 9 [illegible] 12 5
154 DATA 11 0 9 1 8 2 8 5 5 10 13 5
155 DATA 12 1 2 3 [illegible] 11 15
156 DATA 13 1 1 2 1,3 3 6 [illegible] 11 5 16
157 DATA 14,1 2 2 2,3 4 4 8 12 3 17
158 DATA 15,1 3,2,4,3,5 7 1 13 17 5
159 DATA 16 1 3 2 6 3 7 7 3 13 5 21
160 DATA 17,1 4,2 7,3 8,7 5,14 19
161 DATA 18,1 4 2 7 3 9 7 8,14 5 19 8
162 DATA 19,1 5,2 8 4 8 15 20
163 DATA 20,1 5,2 8,4 1,8 1,15 5 21
164 DATA 21,1 5,2 9 4 2 8 2 15 7,21 5
165 DATA 22,1 5,2 9,4 2,8 2 15 7,21 5
166 DATA 23,1 5 2 9 4 3 8 3 15 8 21 7
167 DATA 24 1,5 2 7,4 1 8 1,15 5 21
168 DATA 25,1 5,2 5 4 8,15 20
170 IF v>25 THEN GO TO 115
173 IF v<2 THEN GO TO 110
175 READ V,a,b,c,d,g,h
180 IF v=V THEN GO TO 200
195 GO TO 175
200 IF s=8 THEN LET a=a: LET Nn=a
205 IF s=13 THEN LET b=b: LET Nn=b
210 IF s=17 THEN LET c=c: LET Nn=c
215 IF s=25 THEN LET d=d: LET Nn=d
220 IF s=32 THEN LET g=g: LET Nn=g
225 IF s=40 THEN LET h=h: LET Nn=h
228 CLS
230 LET z=(Db-Da)/(2*Y)
235 LET r=ASN z
240 LET B=(r/PI)*180
245 LET c=180-2*B
250 LET ka=c*0 00375+0 73
255 PRINT AT 5 8;"NAMEN UPORABE"": BEEP 3,3
260 INPUT "Ventilator,Centrifugalna crpalka,
  Transportni trak,Lahki stroj  D/N";o$
270 IF o$(1)="D" OR o$(1)="d" THEN GO TO 310
275 INPUT "Tezki stroj,Batne crpalke,Batni
  motor D/N";o$
280 IF o$(1)="D" OR o$(1)="d" THEN GO TO 315
285 INPUT "Kompresor,Drobilec,Presa,Dvigalo,
  Mehanicne kladivo D/N",o$
290 IF o$(1)="D" OR o$(1)="d" THEN GO TO 320
295 INPUT "Valjcna presa,Bager,Tezka
  strakalnica D/N";o$
300 IF o$(1)="D" OR o$(1)="d" THEN GO TO 325
305 GO TO 260
310 LET k=0.9: GO TO 335
315 LET k=0.8: GO TO 335
320 LET k=0.7: GO TO 335
325 LET k=0.6: GO TO 335
335 LET Nk=k*ka*Nn
340 LET f=N/Nk
341 LET f=INT f
342 IF f<1 THEN LET f=1
343 IF f>12 THEN PRINT AT 5 2;f;"Jermenov na
  pogonu ,priporocljivo do 12 ": BEEP 3,3:
  CLS
345 LET Ca=(B/180)*PI
350 LET C=COS Ca
352 LET v=(((Db*PI)/1000)*nb)/60
355 LET Dj=((Da*PI)/360)*(180-2*B)+((Db*PI)/
  360)*(180+2*B)+(2*Y*C)
356 LET Pa=12*(SQR (SQR ((N*1 36)/na)))*10
358 LET Pb=12*(SQR (SQR ((N*1 36)/nb)))*10
360 LET Dj=Dj-2*s
365 LET G=(2*V)/(Dj/1000)
370 IF G<8 THEN GO TO 380
375 CLS : PRINT  INK 2;AT 3,4;"STEVILO ZVOJEV
  JERMENA": PRINT AT 10,2 G: PRINT AT 10,4,"
  zvojev/sek ear  dovoljeno 20 ": BEEP 1,2:
  PAUSE 200: CLS
```



MERILO=1 1   JERMENICA D=172X78

```
380 PRINT AT 2 3,"PODATKI O POGONU": PRINT AT
  5 0;"Dolzina jermena" Dj: PRINT AT 7,0,"
  Stevilo jermenov ";f: PRINT AT 9,0;"
  Jermenica D1=";Da;" b1=" Pa
381 PRINT AT 11 0; Jermenica D2=";Db; b2= ,Pb:
  PRINT AT 13,0,"Medosna razdalja em ";Y
382 PRINT AT 15,0 Dimenzija jermena: "a:
  PRINT AT 15,20;"X ;Hj
383 PRINT AT 17 0; Hitrost jermena v M/sek ";v:
  BEEP 2 3
385 INPUT  Ali imas ustrezen jermen?D/N ;o$
390 IF o$(1)="D" OR o$(1)="d" THEN GO TO 420
395 INPUT  Ali povecam premer jermenice?D/N";o$
400 IF o$(1)="D  OR o$(1)="d" THEN CLS : GO TO
  110
405 INPUT  Ali povecam medosno razdaljo?D/N";o$
410 IF o$(1)="D  OR o$(1)="d" THEN CLS : GO TO
  120
415 GO TO 5
420 LET Pa=12*(SQR (SQR ((N*1 36)/na))): CLS
425 LET Pa=Pa*10
427 LET h=Hj: LET b=Pa: LET Dg=Da: LET s=a: GO
  TO 460
435 LET Pb=12*(SQR (SQR ((N*1 36)/nb))): CLS
436 LET Pb=Pb*10
450 LET h=Hj: LET b=Pb: LET Dg=Db: LET s=a: GO
  TO 460
460 LET Lj=a*1 5*f
461 LET Dg=INT Dg
495 IF h>110 THEN GO TO 580
545 IF Dg>480 THEN GO TO 580
555 IF Lj>288 THEN GO TO 580
560 IF Dg>240 THEN GO TO 590
565 IF h>55 THEN GO TO 590
570 IF Lj>144 THEN GO TO 590
575 GO TO 640
580 PRINT AT 21 0 "MERILO=1;4": BEEP 3,1:
  PRINT AT 21,24 Dg
583 LET Dg=Dg/2
584 LET Hj=Hj/2
585 LET s=s/2
587 LET b=d/2: GO TO 600
590 PRINT AT 21 0, MERILO=1:2 : BEEP 3,2
  PRINT AT 21 24;Dg
600 LET Dg=Dg/2
601 LET Hj=Hj/2
605 LET s=s/2
610 LET d=d/2: GO TO 1050
640 PRINT AT 21 0  MERILO 1:1": BEEP 3 3:
  PRINT AT 21,24;Dg
1050 LET a=120-(Dg/2)
1051 PRINT AT 21-28 Lj
1055 LET b=Dg
1060 LET j=s/2
1065 LET n=j/2
1070 LET h=Hj*1 25
1073 PLOT 120 30: DRAW 0 40: PLOT 120,75: PLOT
  120,80: DRAW 0 40  PLOT 120,125  PLOT 120,
  130: DRAW 0,40
1075 PLOT a 30  DRAW h 0  DRAW 0,n
1080 LET k=0
1085 LET k=k+1
1090 DRAW  h n: DRAW 0,j: DRAW h,n: DRAW 0,n
1095 IF k<f THEN GO TO 1085
```

```
1100 PLOT a,30: DRAW 0,n
1105 LET 1=0
1110 LET 1=1+1
1115 DRAW h,n: DRAW 0,  DRAW -n,n  DRAW 0,n
1120 IF 1<f THEN GO TO 1110
1135 LET Lr=n+((s+n)*f)
1140 LET m=d/2
1145 LET u=120-(m)
1150 LET r=d*2.5
1160 LET z=Lr-r
1165 LET p=r+30
1170 LET i=m+n
1173 LET x=d/10
1175 LET y=120-(x+m)
1180 IF z<0 THEN GO TO 1210
1185 IF z>0 THEN GO TO 1195
1190 IF z=0 THEN GO TO 1195
1195 PLOT u,30: DRAW 0,r: DRAW m,0: DRAW i,0:
     DRAW 0,z
1200 LET p=Lr+30
1205 PLOT a,p: DRAW b,0: GO TO 1220
1210 PLOT u,30: DRAW 0,r: DRAW m,0: DRAW -i,0:
     DRAW 0,z
1215 PLOT 120,q: DRAW i,0  DRAW 0,z
1220 PLOT y,30  DRAW 0,r
1225 LET k=0
1230 LET w=30
1233 LET lt=Lr+30
1235 LET k=k+1
1240 LET o=t/2 h
1245 LET g=h/2
1250 LET w=w+n: PLOT 120,w: DRAW g,0
1255 LET w=w+n  PLOT 120,w: DRAW o,0
1260 LET w=w+n: PLOT 120,w: DRAW o,0
1265 LET w=w+n  PLOT 120,w: DRAW g,0
1270 IF k<f THEN GO TO 1235
1275 PLOT 120,lt  DRAW g,0
1280 PLOT m,lt
1290 LET to=h/2-1  DRAW to,0
1320 PRINT AT 21,12;"JERMENICA O=" : PRINT AT
     21,27;"X
1350 INPUT "Ali zelis naslednjo skico D/N",c$
1360 IF c$(1)="D" OR c$(1)="d" THEN CLS : GO TO
     435
1370 GO TO 1
```

TELEFOSKI IMENIK

Ko program prepišete, vtipkate telefonske naročnike. Shranite lahko do tisoč imen, spectrum pa v te namen rezervira 15 K pomnilnika. Program posnamete na kaseto in računalniški telefonski imenik je pripravljen.

Dragor Medvešček
Ljubljana

```
2 DATA 2,7, 1,2, 1,2, 2,4, 4,2, 2,6, 2,7
3 CLS
15 DIM z$(900,15)
19 CLS
20 PRINT AT 9,0;"1)ISPIS NAROCNIKOV"
21 PRINT AT 10,0;"2)VPIS NOVIH NAROCNIKOV"
25 PRINT AT 0,5 "TELEFONSKI IMENIK"
30 PRINT AT 11,0;"3)SHRANJEVANJE PODATKOV"
40 RESTORE : FOR i=1 TO 7: READ D,N: BEEP D,N:
   NEXT i
50 INPUT "VSTAVI STEVILKO ";Z
55 IF z>=4 OR z<=0 THEN GO TO 19
60 IF Z=2 THEN GO TO 100
70 IF Z=1 THEN GO TO 500
80 IF Z=3 THEN GO TO 9999
110 FOR x=1 TO 900
120 IF z$(x)="               " THEN GO TO
130 NEXT x
140 STOP
500 CLS
510 INPUT "VSTAVI PRIIMEK";p$
520 INPUT "VSTAVI IME";o$
530 FOR a=1 TO 15
540 LET q=LEN p$
550 IF q=15 THEN GO TO 600
560 LET p$=p$+" "
570 NEXT a
600 FOR a=1 TO 15
610 LET k=LEN o$
620 IF k=15 THEN GO TO 700
630 LET o$=o$+" "
640 NEXT a
700 FOR a=1 TO 900
710 IF p$=z$(a) AND o$=z$(a+1) THEN GO SUB
    5000
720 IF z$(a)="               " THEN GO TO 19
730 NEXT a
5000 CLS
5010 PRINT z$(a)
5020 PRINT z$(a+1)
5030 PRINT z$(a+2)
5040 PRINT "Tel ";z$(a+3)
5050 PRINT AT 21,0;"PRITISNI enter ZA PONOVNO
     IZBIRO
5060 PAUSE 0
5070 CLS
5080 RETURN
8000 CLS
8010 INPUT "VSTAVI PRIIMEK";b$
8020 INPUT "VSTAVI IME";n$
8030 INPUT "VSTAVI ULICO IN HISNO STEVILKO ";s$
8040 INPUT "VSTAVI TEL STEVILKO ";t$
8050 LET z$(x)=b$
8060 LET z$(x+1)=n$
8070 LET z$(x+2)=s$
8080 LET z$(x+3)=t$
8090 GO TO 19
9998 STOP
9999 SAVE "TEL.IMENIK" LINE 19
```

POVECAVA CRK (2)

V drugi številki Mojega mikra sem zasledila program za povečavo črk. Zdi se mi zelo počasen in zaradi izpisa neprimeren za daljše programe. Pošiljam vam svoj program, ki je napisan v strojnem jeziku. Sicer je malce daljši, vendar ima to prednost, da črke lahko premikaš po x in y osi ter določaš barvo teksta; poleg tega program ne poveča samo ene črke, temveč ves tekst. Ima pa to napako, da na koncu vrstice ne skoči v novo, ampak piše čez prejšnjo.
Kratek nasvet: z ukazom MERGE lahko program dodamo daljšim programom.

Marjeta Jurančič
Babelj, Predvor

```
10 GO SUB 2000
20 INPUT "Pomik po X osi (0-200)";xx
30 INPUT "Pomik po Y osi (0-100)";yy
40 INPUT "Faktor za sirino (1-10)";xs
50 INPUT "Faktor za visino (1-10)";ys
55 INPUT "Barva ";col
56 INK col
60 INPUT "Beseda-text";a$
65 INPUT "Razmak med crkami";raz
70 GO SUB 1000
80 PAUSE 0: CLS : GO TO 20
1000 LET i=23305: POKE i,xx: POKE i+1,yy: POKE
     i+2,xs: POKE i+3,ys: POKE i+4,raz: LET i=i+
     4: LET w=LEN a$
1010 FOR n=1 TO w: POKE i+n,CODE a$(n)
1020 POKE i+w+1,255: LET w=USR 32256
1030 RETURN
2000 RESTORE
2010 FOR x=32256 TO 32256+300
2020 READ a
2030 POKE x,a
2040 NEXT x
2050 DATA 33,15,91,124,35,34,0,91,111,60,200,38,
     0,41,41,41,237,75,54,92,9,62,8,50,4,91,58,
     11,91,50,9,91,58,10,91,50,8,91,42,9,50,5,
     91,124,35,34,2,91,7,50,6,91,58,5,91,61,32,
     50,58,4,91,61,32,24,58,14,91,71,58,12,91,
     7,58,10,91,127,5,32,252,50,10
2060 DATA 91,42,0,91,195,3,12,50,4,91,58,13,91,
     71,58,9,91,128,50,9,91,42,2,91,195,32,12,
     50,5,91,58,12,91,71,58,9,91,50,7,91,58,13,
     91,79,197,205,164,126,193,58,7,91,60,50,7,
     91,13,32,241,58,8,91,60,50,8,91,5,32,221,
     58,8,91,195,48,126,126,64,32,16,8,2,1,58,
     142,92,238,255,71,58,141,92
2070 DATA 160,71,58,8,91,230,248,111,58,7,91,
     254,192,208,31,31,31,230,31,103,203,28,203
     29,203,28,203,29,203,28,203,29,62,88,180,
     103,58,142,92,166,176,119,58,7,91,71,230,7
     246,64,103,120,31,31,31,230,24,180,103,120,
     23,23,230,224,111,58,8,91,71,31,31,71,230
     31,181,111,255
2080 DATA 33,156,126,120,230,7,79,60,0,9,70,24
     53,6,81,203,70,40,3,174,18,201,47,174,47,
     18,201,0,0,0,0,0,0,0,0,0,0,0,0,0,23,220
     10,204,11,231,80,26,23,0,0
2090 RETURN
```

KOLEDAR

Program izpiše koledar za poljubnoizbrano leto

Uroš Kastelic
Trebnje

```
10 REM KOLEDAR
20 INPUT "LETO ? ",L
30 LET Z=L+INT ((L-1)/4)+8
40 LET X=Z-7*INT (Z/7)
50 BORDER 6: PAPER 7: INK 0
60 FOR K=1 TO 12
70 CLS : FOR J=1 TO 5: PRINT :
NEXT J: PRINT TAB 14,L
80 PRINT : PRINT
90 READ M$
100 PRINT TAB INT ((32-LEN M$)/2
M$: PRINT
110 READ DM
120 FOR I=1 TO 7
130 PRINT TAB 5,
140 IF I=1 OR I=5 THEN PRINT "P"
;
150 IF I=3 OR I=6 THEN PRINT "S"
;
160 IF I=2 THEN PRINT "T";
170 IF I=4 THEN PRINT "C";
180 IF I=7 THEN PRINT "N";
190 PRINT ,
200 IF K=2 AND L=4*INT (L/4) TH
EN LET DM=29
210 LET A=(I+6-X)-7*INT ((I+6-X
)/7)+1: LET B=0
220 IF A>7-X THEN LET B=B+1
230 GO SUB 400: PRINT TAB 3*B+Y
+8; PRINT A,
240 LET A=A+7: LET B=B+1
250 IF A<=DM THEN GO TO 230
260 IF I=7 THEN LET D=X+DM-7*IN
T (DM/7): LET X=D-7*INT (D/7)
270 PRINT
280 NEXT I: PAUSE 100: NEXT K
290 STOP
300 DATA "JANUAR",31,"FEBRUAR",
28,"MAREC",31,"APRIL",30,"MAJ",3
1,"JUNIJ",30,"JULIJ",31,"AVGUST"
,31,"SEPTEMBER",30,"OKTOBER",31,
"NOVEMBER",30,"DECEMBER",31
400 IF A>9 THEN LET Y=1
410 IF A<10 THEN LET Y=2
420 RETURN
```

SLOVENIJALESLOVENIJALES

programirana prihodnost programirana prihodnost

ISKALEC MIN

Program Iskalec min je pisan za Sharpov PC 1500/1500A Če imate PC 1500, potrebujete še spominsko kartico 4 K

Program startamo z DEF A ali z RUN. Cilj igre je, da ladjo, ki se prikaže na levem koncu LCD zaslona, varno spravimo na desni konec. Računalnik pred začetkom igre postavi na polje 7 nevidnih min. Izslediti jih moramo s čolnom in uničiti z ladijskim topom. Če ladja zadene ob mino, se potopi in igre je konec.

ČOLN : Upravljamo ga s kurzorskima tipkama levo in desno. Boje postavljamo s tipko SPACE Če hočemo ugotoviti oddaljenost čolna od mine, pritisnemo tipko ENTER. Oddaljenost nam prikažeta zvočni signal ali stolpec na desni strani zaslona. Če je čoln nad mino, je zvočni signal najmočnejši, stolpec pa je v tem primeru zapolnjen. Zvočno ali stolpično odkrivanje min menjamo s tipko RCL

LADJA: Upravljamo jo s kurzorskima tipkama levo in desno. S tipko SPACE napolnimo prednji ladijski top. Ko je top napolnjen, spreminjamo njegov doseg s kurzorskima tipkama gor in dol. Top sprožimo s tipko ENTER. Tipka SML aktivira zadnji top. Upravljamo ga tako kot prednji top. Zadnji top je namenjen uničevanju torped.

Tipka [0] je za preklop upravljanja LADJA-ČOLN

Pojasnitev spremenljivk:

-grafične spremenljivke:

- -A$(0) : detonacija mine v drugem delu
- -B$(0) : ladja
- -C$(0) : čoln
- -D$(0) : detonacija mine v prvem delu
- -E$(0) : eksplozija mine ob torpedo
- -F$: boja
- -G$: torpedo
- -H$: periskop podmornice
- -I$: velika granata
- -K$: indikator
- -L$: mala granata
- -M$: simbol za čoln
- -N$: simbol za čoln
- -O$: simbol slušalk (zvočno odkrivanje min)
- -S$: detonacija granate
- -F$: potujoči torpedo
- -Z$(0 7): polnitev stolpca (stolpično odkrivanje min)
- -MX : invertiranje zaslona
- -HW : lok granate izstreljene iz topa

-druge spremenljivke:

- -A : položaj zadnjega konca ladje
- -C : 1=top poln / 0=top prazen
- -DEFGHIJ : položaj min
- -K : položaj zadnjega konca čolna
- -L : dolžina strela
- -M : ASCII vrednost tipk INKEY$
- -N : 1=upravljanje ladje 0=upravljanje čolna
- -P : 1=veliki top , 0=mali top
- -T : T=1 ladja je zapeljala na mino
- -U : položaj boje
- -V : zastreljena mina
- -X : 1=čoln potopljen / 0=čoln OK
- -Z : položaj zadnjega konca torpeda
- -KG : izstreljena velika granata
- -KL : izstreljena mala granata
- -SW : dolžina strela (od GCURSOR 0)
- -TP : uničena torpeda
- -XC : 1=postavljena boja

```
5 REM A PREZELJ
10. A'CLS !CLEAR
   WAIT 30:PRINT
      * ISKALE
   C ** MIN *'
   RANDOM
20.DIM B$(0)*56,A
   $(0)*22, C$(0)*
   24, D$(0)*10, E$
   (0)*38, Z$(7)*1
   2
25 Z$(7)- 0000000
   00000  Z$(6)=
   404040404040'
   Z$(5)='5060506
   06050'
26 Z$(4)='7070707
   07070' Z$(3)-'
   787070787070
   Z$(2)='7C7C7C7
   C7C7C'
27 Z$(1)='7E7E7E7
   E7E7E .Z$(0)-
   7F7F7F7F7F7F
30.B$(0)- 005A0C4
   C5C4E5B405E4A5
   H405A4E5C4C5C4
   05C4C5E4E4E2A1
   A0A02
35 M$='003C220C00
   00  L$='407074
   704000  S$-'40
   5264101064524
40 A$(0)- 1000140
   52976290514001
   0  C$(0)='1030
   5C545C5050381
   40704 0'
50 D$(0)- 2010102
   04020101020  E
   $(0)= 20102503
   1E214E717E707E
   714E211E092510
   20
55 F$='4C6C7C6040
   N$='7C424141
   427C  O$='307E
   01017E30'
60 G$= 4050706040
   4040  H$='7070
   103030  I$= 40
   7E70707E40'
70 A-0 C=0.N=1 I=
   500 J=500 K=30
   L-10 UB$= ' F
   I$='  U-0
80 D=40+RND 70.E=
   40+RND 35 F=40
   +RND 61 G=40+
   RND 67.H=40+
   RND 72 R=0 Y-0
   0
90 IF G>80LET I=4
   0+RND 39
100 IF H<60LET J=4
   0+RND 70
105 CLS  WAIT 0
110 GCURSOR A
   GPRINT B$(0),
   GCURSOR K
   GPRINT C$(0),
   GCURSOR 120
   GPRINT '7E7E',
    CURSOR 21
   PRINT STR$ L,
120 M-3 N 1.X=0
   GOTO 240
130 IF A>=55AND
   RND 70>20AND R
   =0LET DD-0
   GOSUB 1030
131 IF R=1AND RND
   30>5LET ZZ=0
   GOSUB 1050
132 M=ASC INKEY$ :
   ZZ=ZZ+1.DD=DD+
   1
135.IF M 0GOTO (13
   2-(ZZ>=5)-(DD>
   =30))
140.IF M-11AND C=1
   AND L<34AND N=
   1WAIT 0 L=L+1+
   4*(P=0).CURSOR
   21 PRINT     '
   CURSOR 21
   PRINT STR$ L,
   GOTO 130
150.IF M=10AND C=1
   AND L>10AND N=
   1WAIT 0.L=L-1-
   4*(P=0) CURSOR
   21 PRINT     '
   CURSOR 21
   PRINT STR$ L,
   GOTO 130
160 IF M=12AND N=1
   AND A<YGOSUB 2
   72 WAIT 0
   GCURSOR ABS A
   GPRINT  00 , A
   =A+1 GCURSOR A
   .GPRINT B$(0),
   GOTO 280
170 IF M=12AND N=0
   AND K<100AND X
   =0WAIT 0
   GCURSOR K
   GPRINT  00 , K
   -K+1 GCURSOR K
   GPRINT C$(0),
   GOTO 130
178 IF M=13AND X-0
   AND N=0THEN 71
   0
182 IF M=8AND N=0
   AND K>A+27AND
   X-0WAIT 0
   GCURSOR (K+11)
   GPRINT  00 ,
   K=K-1 GCURSOR
   K.GPRINT C$(0)
   , GOTO 360
190 IF M=8AND N=1
   AND A>0WAIT 0.
   O-POINT (A+26)
   .GCURSOR (A+26
   ) GPRINT O 2*(
   O>=2), A=A 1-M
   X=1
200 IF MX=1LET MX=
   0.GCURSOR A
   GPRINT B$(0),
   GOTO 300
210.IF M-13AND N=1
   AND C-1LET C=0
   K$=0$(13+N) K
   G=KG+(P-1) KL=
   KL+(P=0) GOTO
   390
220 IF M=2AND N-1
   AND A>-45LET C
   =1 P=0 K$ L$
   WAIT 0.GCURSOR
   147 GPRINT K$,
   GOTO 130
225 IF M=32AND X=0
   AND N-0AND K>A
   +34AND XC=0LET
   U=K-5 BEEP 1,5
   0,50.WAIT 0.
   GCURSOR U
   GPRINT F$, XC=
   1 GOTO 130
230.IF M=32AND N=1
   LET P=1 C=1 K$
   =I$ WAIT 0
   GCURSOR 147.
   GPRINT K$,
   GOTO 130
240.IF M=9AND N=1
   AND X=0LET N-0
   C=0 K$=M$ P=1
   GCURSOR 147
   WAIT 0 GPRINT
   K$, GOTO 130
250.IF M=9AND N-0
   LET N=1 C=0 K$
   =N$.P=1
   GCURSOR 147.
   WAIT 0.GPRINT
   K$, GOTO 130
260 IF M=25AND N=0
   AND X-0AND W=0
   LET K$-O$ W=1
   WAIT 0.GCURSOR
   147 GPRINT K$,
   GOTO 130
265 IF M=25AND N=0
   AND X=0AND W=1
   LET W=0 WAIT 0
   GCURSOR 147
   GPRINT  787E78
   738443 , GOTO
   130
270.GOTO 130
272 IF A>=U 27AND
   U>0LET A=A-1
   RETURN
273 IF A>=K-27AND
   K>0LET A=A-1
   RETURN
274 RETURN
280 FOR S-4TO 10.
   IF 0(S)=A+23
   LET 0(S)-300 T
   =1E6
290 NEXT S
300 IF T 1E6THEN 0
   10
310 IF A-YTHEN 310
320.IF A+25>=KAND
   X-0LET X=1 Y-9
   3 K=0
340 GOTO 130
350 BEEP 1,60,3.
   WAIT 0 GCURSOR
   U GPRINT '0000
   000000
   GCURSOR A
   GPRINT B$(0),
   U-0 XC=0 GOTO
   130
360 IF K<U+5WAIT 0
   GCURSOR U XC=
   0 GPRINT  0000
   000000 ;
   GCURSOR K
   GPRINT C$(0),
   GCURSOR A
   GPRINT B$(0),
   U 0
370 GOTO 130
380 IF A+1<=Z+7AND
   F1$=G$LET T=1E
   -3 GOTO 820
390 GOTO 130
395 WAIT 0 BEEP 1,
   150,50 GCURSOR
   147.GPRINT K$,
400 IF P=0AND A>-4
   5THEN 600
410.SW=A+L+27 IF S
   W>=120LET SW-1
   20
420 FOR S=A+29TO S
   W-3.HW=2+POINT
   S.GCURSOR S
   GPRINT HW,
   NEXT S
430 GCURSOR (SW-2)
   .HW=POINT (SW-
   2)+4 GPRINT HW
   , GCURSOR (SW-
   1) HW=POINT (S
   W-1)+8
440 GPRINT HW,
   GCURSOR SW HW=
   112+POINT SW
   WAIT 30 GPRINT
   HW,
450.GOSUB 500
490 IF U>0AND SW>=
   U 1AND SW<=U+5
   WAIT 50.BEEP 2
   ,150,30 U=0.XC
   =0 V=V+1
   GCURSOR SW-3
   GPRINT S$,
   GOTO 520
500.IF X-0AND SW>=
   KAND SW<=K+12
   WAIT 50 BEEP 4
   ,150,30.X-1.
   GCURSOR SW 3
   GPRINT S$,
   GOTO 790
510.WAIT 30
   GCURSOR SW-4
   BEEP 1,50,15
   GPRINT O$(0),
   BEEP 1,70,20
   GCURSOR SW-5.
   GPRINT A$(0),
520 HW=0 XM=0 ZM=0
   FOR S=4TO 10
   IF (SW>=0(S)-1
   )*(SW<-0(S)+1)
   LET 0(S)=0 ZM-
   ZM+1 XM-1
525 IF ZM=1AND XM-
   1LET HW-1 V=V+
   1
527 XM-0
530 NEXT S
540 IF HW-1LET HW-
   0 BEEP 5,5,5
   WAIT 50
   GCURSOR SW 3.
   GPRINT E$(0),
   GOTO 550
```

```
550 GCURSOR SW-5
    WAIT 50 BEEP 6
    ,5,7 GPRINT A$
    (0),
555 GOSUB 560.GOTO
    130
560.WAIT 0 CLS
    GCURSOR 2
    GPRINT FI$,
    GCURSOR A
    GPRINT B$(0),
570 IF U>0GCURSOR
    U GPRINT F$,
580 IF X=0GCURSOR
    K.GPRINT C$(0)
    ,
590 GCURSOR 120.
    GPRINT 7E7E ,
     CURSOR 21.
    PRINT STR$ L,
    GCURSOR 147
    GPRINT K$,
    RETURN
600.SW-A+1 L IF SW
    <=9LET SW-9
610 WAIT 0 FOR S=A
    -1TO SW+5STEP
    -1
620 GCURSOR S HW=1
    +POINT S
    GPRINT HW,
    NEXT S
630.GCURSOR SW+4 H
    W=2+POINT (SW+
    4  GPRINT HW,
    GCURSOR SW+3 H
    W-4+POINT (SW+
    3) GPRINT HW,
640 GCURSOR SW+2 H
    W=8+POINT (SW+
    2) GPRINT HW,
    GCURSOR SW+1 H
    W=16+POINT (SW
    +1) GPRINT HW,
650 GCURSOR SW HW=
    56+POINT SW
    WAIT 30 GPRINT
    HW, GOSUB 560
660 IF SW>-ZAND SW
    <=Z+7THEN 650
670 WAIT 30 BEEP 1
    ,50,15 GCURSOR
    SW-4 GPRINT D$
    (0),
682 GCURSOR SW-5
    BEEP 1,70,20
    GPRINT A$(0),
    GOSUB 560 WAIT
    0 C-1 P=0-
    GCURSOR 147
    GPRINT L$,
    GOTO 130
690.WAIT 30.BEEP 2
    ,150,30
    GCURSOR SW-3
    GPRINT S$ BEEP
    5,5,5 WAIT 50
    GCURSOR SW-9
    GPRINT E$(0);
700.TP=TP+1 R=0.FI
    $- '-GOSUB 560
    WAIT 0.C=1 P=
    0.GCURSOR 147
    GPRINT L$, Z=0
    GOTO 130
710 IF W-1THEN 780
715 GOSUB 720 GOTO
    770
720.HW=0
740 IF ABS (K+10-D
    )-HWOR ABS (K+
    10-E)=HWOR ABS
    (K+10-F)-HWOR
    ABS (K+10-G)=H
    WRETURN
750 IF ABS (K+10-H
    )=HWOR ABS (K+
    10-I)=HWOR ABS
    (K+10 J) HW
    RETURN
760 HW=HW+1 IF HW<
    =7THEN 740
765 HW=7 RETURN
770 WAIT 0 GCURSOR
    147 GPRINT Z$(
    HW), GOTO 130
780 GOSUB 720 HW-7
    -HW BEEP HW,5,
    50 GOTO 130
790 BEEP 5,30,10.
    GCURSOR K WAIT
    30.GPRINT '404
    0705070600040606
    02060 , Y=93
800 GCURSOR K
    GPRINT O$(0),
    K-0.GOSUB 560.
    GOTO 130
810 GCURSOR (A+14)
    WAIT 60 BEEP
    5,100,2 GPRINT
    E$(0), GCURSOR
    (A+27) WAIT 0
    GPRINT 000000
    000000 ,
    GCURSOR A
    GPRINT B$(0),
    IF X=0GCURSOR
    K GPRINT C$(0)
    ,
816 IF XC 1GCURSOR
    U.GPRINT F$,
817 GOTO 835
820 GCURSOR (Z+3)
    WAIT 60 BEEP 5
    ,00,2 GPRINT
    E$(0),
835 FOR S-2TO 6
    STEP 2 RESTORE
    (900+S) READ 0
    $(0)
840 WAIT 0 CURSOR
    21.PRINT '
     ',
850 WAIT 5 CURSOR
    22 BEEP 3,20,1
    50 PRINT  S ,
    BEEP 3 20,400
    PRINT  O ,
    BEEP 3,20,150
    PRINT  S ,
860 WAIT 40
    GCURSOR A
    GPRINT B$(0),
    NEXT S
870 WAIT 0 GCURSOR
    A+5.GPRINT '40
    2020406010121 1
    61422042611112
    106040202040'
880.CURSOR 22
    PRINT      ;
    CURSOR 23.WAIT
    120.GPRINT '40
    22223F222240
    GOTO 1E3
902 DATA  2060303 0
    7020602070286C
    2C6838703070 20
    7030703030286 8
    2800'
904 DATA '0020404 0
    4000000060203 0
    3020604040400 0
    4040600060202 0
    2020
906 DATA  0000000 0
    000000020200 40
    4000000000000 2
    0000000000000 0
    0000
910 BEEP 1,5,50
    BEEP 1,70,50
    BEEP 1 150,90.
    BEEP 1,150,100
     BEEP 1,60,60.
    BEEP 1,200,200
920 WAIT 35
    GCURSOR 147
    GPRINT  183020
    657F65203018',
930 WAIT 95 CLS
    PRINT        PO
    RABA MATERIALA
    ,
931 WAIT 0 FOR MZ-
    15TO 139'MX-
    POINT MZ
933 MX-127-MX.
    GCURSOR MZ
    BEEP 1,1,1.
    GPRINT ABS MX,
    NEXT MZ
935 MX=127-POINT 1
    40 GCURSOR 140
    WAIT 80
    GPRINT ABS MX
    CLS
940 WAIT 0 PRINT
    SOUR   ,
    CURSOR 10
    GPRINT G$,
    CURSOR 12
    PRINT TP,
950 CURSOR 20.
    GPRINT  08805D
    222A22500808 ,
     CURSOR 22
    WAIT  PRINT U
960.WAIT 0 PRINT
    TVOJE
    CURSOR 7
    GPRINT C$(0),
    CURSOR 3 PRINT
    X,
970 CURSOR 14
    GPRINT I$,
    CURSOR 16
    PRINT KG,
    CURSOR 21
    GPRINT L$
    CURSOR 23 WAIT
    PRINT KL,
975 CLS  S=(10*TP+
    5*U)/(20*X+2*K
    G+KL)
980 WAIT  IF S>1
    LET S=INT (10*
    S)/10.PRINT  S
    OUR  IZGUBE ,
    S, *VECJE  .
    GOTO 1E3
990.IF S<1LET S-1/
    S S=INT (10*S)
    /10.PRINT  TUO
    JE IZGUBE  ;S;
     * VECJE '
1000 CLS  INPUT
    se ena  gro
    ? (DA/NE) ,U
    $ IF U$- DA
    THEN 10
1010 END
1030 R-1 WAIT 80
    GCURSOR 5
    GPRINT H$,
    WAIT 0
    GCURSOR 5
    GPRINT  P000
    00000P
1040 Z RND 10+5
    GCURSOR Z.
    GPRINT S$,
    R,TURN
1050 WAIT 0
    GCURSOR Z
    GPRINT  000A
    000000000 ,
     Z=Z+3+RND 1
    P
1060 GCURSOR Z
    GPRINT G$, F
    I$ G$
1070 IF A+I<  F7
    AND F1$ G$
    LET T E 9-
    WAIT 0
    GCURSOR Z
    GPRINT  0000
    00000000
    GOTO 820
1080 RETURN
```

BITNI VZOREC

Pri programiranju in v šolah moramo mnogokrat pretvarjati ali kombinirati številčne sisteme. Tu vedno nastopajo dvojiški, desetiški in šestnajstiški (binarni, decimalni in heksadecimalni) sistemi. Tako si moramo pri programiranju v strojnem jeziku ali drugih posegih v pomnilnik računalnika vedno ustvarjati sliko pomnilniške enote, glede na vklop in izklop posameznih bitov. Največkrat si pomagamo z risanjem bitnega vzorca.

Program nam daje pripomoček za preračunavanje vrednosti desetiških števil v dvojiška in šestnajstiških števil v desetiška in dvojiška. Pri tem nam vsakokrat nariše bitni vzorec zadnjega byta. Kadar vnašamo vrednosti večje od 255 oziroma od $FF, nam pokaže še vrednost Hb. To je vrednost naslednjega byta. Pozneje pa mnogokratnik števila 256 pri številih, ki jih ne moremo zadelati v enojnem bytu.

Če na primer vnesemo desetiško število 49356 nam program pokaže naslednje:

Hb=192
Lb=204 in bitni vzorec
1 1 0 0 1 1 0 0
|_|_|_|_|_|_|_|_|

Če vnesemo na primer šestnajstiško število $COCF, nam najprej pokaže desetiško vrednost števila nato pa enako naprej kot pri pretvorbi desetiškega v dvojiško število. Kakor druge vrednosti nam obdrži s oznako na zaslonu tudi šestnajstiško število, ki ga lahko torej primerjamo z bitnim vzorcem. To koristi predvsem tistim, ki se šele uvajajo v različne sisteme.

Program lahko torej uporabljamo večstransko, predvsem pa v šolah, kjer je takšno hitro pretvarjanje z vizualnim prikazom dobro sredstvo.

Program je sestavljen takole:

Vrsta 5-142 oblikujejo napise na zaslonu in dajo izhodišče menu za izbor desetiškega oziroma šestnajstiškega vnosa.
Vrsta 144 zahteva, da pritisnemo D ali H, na druge ne reagira.
Vrsta 146 nas od izboru H preusmeri k podprogramu od 500 do 595.
Vrsta 160 nam izpiše desetiško število, tudi če smo ga vnesli šestnajstiško.
Vrsta 165-240 preračunajo desetiško število v dvojiško in jih vežejo na posamezne spremenljivke glede na položaj bitov.
Vrsta 250-300 izpišejo rezultat v določeni grafični obliki.
Vrsta 310 daje možnost za ponoven vnos ali končanje programa.
Vrsta 320-350 zahtevajo pritisk tipke D ali N in izvedejo ukaz le s tema znakoma.
Vrsta 500-595 so podprogram za preračun šestnajstiškega števila v desetiško. Ta del zahteva vnos le veljavnih znakov šestnajstiškega sistema, torej 0-9 in A-F. Na drugačne znake ne reagira.
Program je sicer pisan za commodore, vendar ga je zelo enostavno prirediti tudi za spectrum z vnosom nekaterih ukazov, npr. LET pred spremenljivkami.

Stane Fele
Ljubljana

SLOVENIJALESSLOVENIJALES

programirana prihodnost programirana prihodnost

```
9 PRINT"♥"
10 REM ******************
20 REM *                *
30 REM *   BITNI VZOREC  *
40 REM *                *
50 REM ******************
60 PRINT "♥"
70 PRINT "█            █BITNI VZOREC█"
80 REM ZA SAVE! "BIT"
90 PRINT "█           DEKADNEGA STEVILA"
100 PRINT
110 PRINT " VSTAVI STEVILO D "
115 PRINT
142 PRINT"ZELIS VNASATI DEKADNO ALI HEX ? (D/H)"
144 GET A$:IF A$="" OR A$<>"D" AND A$<>"H" THEN 144
145 PRINT
146 IF A$="H" THEN 500
150 INPUT " D=";D
155 PRINT
160 PRINT" D="D
165 M=INT(D/256):N=D-256*M
170 H=INT(N/2↑7)
180 G=INT(N/2↑6-2↑1*H)
190 F=INT(N/2↑5-2↑2*H-2↑1*G)
200 E=INT(N/2↑4-2↑3*H-2↑2*G-2↑1*F)
210 D=INT(N/2↑3-2↑4*H-2↑3*G-2↑2*F-2↑1*E)
220 C=INT(N/2↑2-2↑5*H-2↑4*G-2↑3*F-2↑2*E-2↑1*D)
230 B=INT(N/2↑1-2↑6*H-2↑5*G-2↑4*F-2↑3*E-2↑2*D-2↑1*C)
240 A=INT(N/2↑0-2↑7*H-2↑6*G-2↑5*F-2↑4*E-2↑3*D-2↑2*C-2↑1*B)
250 PRINT"▯▯▯▯▯"
260 PRINT"█"
270 PRINT
275 PRINT TAB(13);"┌┐┌┐┌┐┌┐┌┐┌┐┌┐┌┐"
280 PRINT"STEV.BITA";TAB(13);"|7||6||5||4||3||2||1||0|"
285 PRINT TAB(13);"└┘└┘└┘└┘└┘└┘└┘└┘"
290 PRINT" HB="M
295 PRINT "█ LB=" N TAB(13);H;G;F;E;D;C;B;A
300 PRINT TAB(14);"└─┴─┴─┴─┴─┴─┴─┘"
305 PRINT
310 PRINT" ZELIS NASLEDNJO STEVILO? (D/N)"
315 PRINT
320 GET A$:IF A$=""OR A$<>"D" AND A$<>"N" THEN320
330 IF A$="N" THEN 350
340 GOTO 5
350 PRINT"     HVALA ZA SODELOVANJE!":END
500 REM ** HEX TRANS **
502 INPUT"VNESI STIRIMESTNO HEX STEV ";X$
505 PRINT TAB(26);"┴┴┴┴"
510 FORX=4 TO 1 STEP -1
515 G$=MID$(X$,5-X,1)
520 IF ASC(G$)=42 THEN END
525 IF ASC(G$)>57 AND ASC(G$)<71 THEN545
530 IF ASC(G$)<48 OR ABC(G$)>70 THEN500
535 G=ASC(G$)-48
540 GOTO 550
545 G=ASC(G$)-55
550 D=D+G*16↑(X-1)
555 NEXT
595 GOTO 155
```

JACKPOT

Igra gotovo poznate, lasiv se ta program ga ja bila objava podobnega za ZX 81. Pisan je za commodore 64.

Marjan Trusi
Walla Walla
Washington

```
1 PRINT"♥" PRINT PRINT PRINT PRINT PRINT PRINT PRINT TAB(10) TRUEBOFT - 1984
2 PRINT TAB(10) "****************" FOR N=0 TO 4000 NEXT
5 PRINT"♥" POKE53280,0 POKE53281,0 POKE52 48 POKE56 48 CLR GOSUB600
10 [illegible]
15 POKES+1 112 POKES+5 9 POKES+15 200 POKES+24 15
20 POKES+8 150 POKES+12 0 POKES+13 0
25 PRINT"♥" FOR A=1105 FORB=11040 PRINT █ NEXTB A
30 FORB=1105 PRINT █  █  █ █  █ █  █  NEXT
35 FORB=1105 FORB=11040 PRINT █ NEXTB A POKE53269 255
40 PRINT PRINT [illegible]
45 PRINT PRINT [illegible]
50 PRINT PRINT [illegible]
55 PRINT PRINT [illegible] GOSUB400
60 POKE53269 1 POKE 52 255 DIM [illegible]
65 PRINT [illegible] JACKPOT
70 PRINT [illegible] PRIDEL BOS Z 50 DINARJI ZA POSKUSAL
75 PRINT [illegible] TER UMAKNIL ZA SRECO
80 PRINT [illegible] O JE BO STALO 1 DINAR ZA VSAK POSKUS!
85 PRINT"[illegible] UPORABLJAJ TIPKE 1-4
90 PRINT [illegible] VISJA KOT JE STEVILKA TEZJE JE DOBITI
95 PRINT"[illegible]
100 PRINT [illegible] CE ZELIS PREKINITI [illegible]
105 PRINT [illegible] PRITISNI [illegible]
110 DIM WIN$(24)
115 FORA=11024 READWIN$(A) NEXT
120 DATALEMON,CHERRY,CHERRY 5 LIME,LIME,LIME 10 [illegible]
125 DATAPLUM,PLUM,PLUM 14 [illegible]
130 DATABELL,BELL,BELL 18 SEVEN,SEVEN,SEVEN 50 BAR,BAR,BAR 100
135 DATALEMON,LEMON,LEMON 18 CHERRY,CHERRY,CHERRY 15
140 [illegible]
145 [illegible]
150 [illegible]
155 DATA 2 3 7 5 6 7 4 3 1 5 5 7 4 6 7 6
160 DATA [illegible]
165 DATA 2 3 4 5 [illegible]
170 FORI=[illegible]
175 [illegible]
180 [illegible]
185 [illegible]
190 NEXT
195 [illegible]
200 [illegible]
205 [illegible]
210 [illegible]
215 POKE53249 40 POKE53251 [illegible]
220 POKE53257 40 [illegible]
225 [illegible]
230 [illegible]
235 [illegible]
240 [illegible]
245 PRINT [illegible]
250 [illegible]
255 PRINT [illegible]
260 [illegible]
265 ON[illegible] 270 275 280 285 290 295
270 NEXT [illegible]
275 [illegible]
280 [illegible]
285 NEXT [illegible]
290 [illegible]
295 [illegible]
300 [illegible]
305 [illegible]
310 [illegible]
315 [illegible]
320 [illegible]
325 [illegible]
330 [illegible]
335 [illegible]
340 [illegible]
345 [illegible]
350 [illegible]
355 [illegible] END
360 [illegible]
365 SPIN$=SPIN$+ BAR RETURN
370 SPIN$=SPIN$+ SEVEN RETURN
375 SPIN$=SPIN$+ BELL RETURN
380 SPIN$=SPIN$+ CHERRY RETURN
385 SPIN$=SPIN$+ LIME RETURN
390 SPIN$=SPIN$+ PLUM RETURN
395 SPIN$=SPIN$+ LEMON RETURN
400 [illegible]
405 [illegible]
410 [illegible]
415 [illegible]
420 DATA [illegible]
425 [illegible]
430 [illegible]
435 [illegible]
440 [illegible]
445 [illegible]
450 [illegible]
455 [illegible]
460 [illegible]
465 [illegible]
470 [illegible]
475 [illegible]
480 [illegible]
485 [illegible]
490 [illegible]
495 [illegible]
500 [illegible]
505 [illegible]
510 [illegible]
515 [illegible]
520 [illegible]
525 [illegible]
530 [illegible]
535 [illegible]
540 [illegible]
545 [illegible]
550 [illegible]
555 [illegible]
560 [illegible]
```

BIOROBOT

Na začetku igre na commodore 64 ste v središču. Neprestano vas napadajo tako imenovani "robotki", ki puščajo za sabo bioboobe. Če vas kateri ujame, če naletite na bombo ali se zaletite v zid visoke napetosti, izgubite eno od življenj. Igra se z igralno palico v vratih 2. Streljate tako, da pritisnete gumb FIRE in smer streljanja, za vsak uničen biorobot dobite 10 točk. Na 200 in 800 točk dobite dodatno življenje in 100 točk. Program je v basicu, razen najhitrejše dele v strojnem jeziku za neprestano spreminjanje barv.

Pavel Škerlj
Tomaj

[illegible]

SLOVENIJALESLOVENIJALES

programirana prihodnost programirana prihodnost

SEZNAM

Se vam je že zgodilo, da ste izgubili pregled nad vsebino svojih neštetih diskov? Potem je zadnji čas, da z naslednjim programom naredite red v svoji "disketeki".

Program SEZNAM zapiše na zaslon in tiska v petih stolpcih zelo pregleden seznam zapisov na diskih.

Napisan je za računalnik commodore 64, disketno enoto 1541 in tiskalnik FX-80. Ker je pisan v basicu, ga lahko z majhnimi spremembami prilagodite tudi drugim računalnikom, disketnim enotam in tiskalnikom.

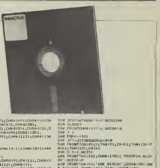
Tiskalnik FX-80 je priključen neposredno po kablu na uporabnikova vrata in razen software interfacea ne potrebuje nikakršnega vmesnika.

Program deluje brezhibno z različnimi vmesniki, npr.:

1.1 interface podjetja Eisele (opisan v reviji 64'er, 7/84). Naslov tiskalnika 16.

2.1 Software interface podjetja Data Becker (opisan v knjigi "64-Tips & Tricks"). Naslov tiskalnika je 2, sek. naslov je 1. Naslov tiskalnika je treba v vrstici 120 ustrezno spremeniti.

Janis Arko
Northese



```
100 REM *SEZNAM ZAPISOV NA DISKU* V1/14.
10.84
110 DIMZ$(144),F$(144),E$(144)
120 OPEN1,16
130 PRINT#1,CHR$(27),CHR$(78),CHR$(7);
140 PRINT#1,CHR$(27);CHR$(82);CHR$(0);
150 PRINTCHR$(147);:PRINT:PRINT:PRINT:PR
INTTAB(6);
160 INPUT"NAPISI DATUM";T$
170 PRINTCHR$(147);:PRINT:PRINT:PRINT
180 PRINTTAB(8),"JE DISK VLOZEN? (D/N)"
190 GETA$:IFA$=""THEN190
200 IFA$<>"N"THEN220
210 IFA$="N"THEN180
220 PRINTCHR$(147);
230 FORI=1TO6:PRINT:NEXT:PRINTTAB(14);"*
PODATKI*"
240 FORI=1TO144:Z$(I)="":F$(I)="":E$(I)=
"":NEXT
250 OPEN15,8,15,"I"
260 INPUT#15,EN,EM$,ET,ES:IFEN=0THENCLOS
E15:GOTO320
270 PRINTCHR$(147):PRINT:PRINT:PRINT
280 PRINTTAB(11);"NAPAKA NA DISKU ":PRIN
T:PRINT
290 PRINTTAB(2);"STEV. NAPAKE:",EN,"=";E
M$:PRINT
300 PRINTTAB(2);"TRAK.";ET;SPC(4);"SEKTO
R",ES:PRINT:PRINT
310 CLOSE15:GOTO1000
320 N=0:G=0:T=1:OPEN2,8,0,"$0"
330 GET#2,A$,B$
340 GET#2,A$,B$
350 GET#2,A$,B$
360 C=0:S$="":V$="":M$="":W=W+1:G=G+1
370 IFA$<>""THENC=ASC(A$)
380 IFB$<>""THENC=C+ASC(B$)*256
390 V$=MID$(STR$(C),2)
400 FORJ=LEN(V$)TO3:V$=" "+V$:NEXTJ
410 Z$(I)=RIGHT$(V$,3)
420 GET#2,B$:IFST=64THEN530
430 IFB$<>CHR$(34)THEN420
440 GET#2,B$:IFB$<>CHR$(34)THENS$=S$+B$:
GOTO440
450 F$(I)=S$
460 GET#2,B$:IFB$=CHR$(32)THEN460
470 C$=""
480 C$=C$+B$:GET#2,B$:IFB$<>""THEN480
490 E$(I)=LEFT$(C$,2)
500 IFW=1THENGOTO640
510 E$(I)=LEFT$(E$(I),1)
520 IFST=0THENI=I+1:GOTO340
530 CLOSE2
540 PRINTCHR$(147):GOTO410
550 T=1
560 FORK=1TOI
570 IFT=22THENGOSUB1070
580 PRINTTAB(4);K;TAB(9);Z$(K);TAB(15);F
$(K);TAB(32);E$(K)
590 T=T+1:NEXTK
600 PRINTTAB(15);CHR$(145);"PROSTIH BLOK
OV":GOTO660
610 PRINTTAB(4);"IME DISKA:";CHR$(18);DN
$,TAB(27),CHR$(146),"ID=";CHR$(18);
620 PRINTID$;CHR$(157),CHR$(146):PRINT
630 GOTO550
640 DN$=F$(1):ID$=E$(1):I=0:E$(1)=LEFT$(
E$(1),2)
650 GOTO520
660 PRINT:PRINTTAB(9);"TISKATI SEZNAM? (
D/N)"
670 GETA$:IFA$=""THEN670
680 IFA$<>"N"THEN700
690 IFA$="N"THEN1000
700 N1=(K-1)/5
710 IF N1>INT(N1)THENN=INT(N1)+1
720 IFN1<1ORN1=1THENN=1
730 IFN1=INT(N1)THENN=N1
740 F$(K-1)="PROSTIH BLOKOV":E$(K-1)="*"
750 PRINT#1,CHR$(27);CHR$(33);CHR$(0);
760 PRINT#1,TAB(8);"DISK:";
770 PRINT#1,CHR$(27);CHR$(33);CHR$(8);
780 PRINT#1,SPC(1);DN$;SPC(1);
790 PRINT#1,CHR$(27);CHR$(33);CHR$(0);
800 PRINT#1,"ID",
810 PRINT#1,CHR$(27);CHR$(33);CHR$(8);
820 PRINT#1,ID$,SPC(30),T$
830 PRINT#1,CHR$(27),CHR$(33);CHR$(0);
840 PRINT#1,CHR$(27),CHR$(15);
850 PRINT#1,CHR$(27);CHR$(68);CHR$(9);CH
R$(13),CHR$(30),CHR$(39);
860 PRINT#1,CHR$(55),CHR$(55),CHR$(63),C
HR$(80),CHR$(84),CHR$(105),
870 PRINT#1,CHR$(109),CHR$(113),CHR$(130
);CHR$(0);
880 FOR I1=1TON
890 I2=I1+N:I3=I1+2*N:I4=I1+3*N:I5=I1+4*
N
900 PRINT#1,CHR$(9);
910 PRINT#1,Z$(I1);CHR$(9);F$(I1),CHR$(9
),E$(I1),CHR$(9);Z$(I2);CHR$(9);
920 PRINT#1,F$(I2);CHR$(9);E$(I2);CHR$(9
),Z$(I3);CHR$(9);F$(I3);CHR$(9);
930 PRINT#1,E$(I3),
940 PRINT#1,CHR$(9);Z$(I4);CHR$(9);F$(I4
),CHR$(9),E$(I4);CHR$(9);Z$(I5);
950 PRINT#1,CHR$(9);F$(I5);CHR$(9);E$(I5
)
960 NEXTI1
970 PRINT#1,TAB(9);
980 FORI=1TO122:PRINT#1,"-";:NEXT
990 PRINT#1
1000 PRINT:PRINT:PRINTTAB(10);"SE EN DIS
K? (D/N)"
1010 GETA$:IFA$=""THEN1010
1020 IFA$<>"N"THEN170
1030 PRINTCHR$(147);:FORI=1TO7:PRINT:NEX
T
1040 PRINTTAB(11),"*KONEC PROGRAMA*"
1050 PRINT#1,CHR$(27);CHR$(33);CHR$(0),
1060 CLOSE1:END
1070 PRINTTAB(10),CHR$(17);"NAPREJ Z "[C
HR$(18)," SPACE ";CHR$(146)
1080 GETQ$:IFQ$=""THEN1080
1090 IFQ$=" "THENPRINTCHR$(147);:T=0:RET
URN
1100 IFQ$<>" "THEN1080
```

DISK. 1541TEST/DEMO ID=ZX 15.10.84

13	HOW TO USE	P	4	DOS 5.1	P	4	DIR	P	9	PERFORMANCE TEST	P
5	HOW PART ONE	P	11	COPY/ALL	P	6	VIEW BAM	P	5	SEQUENTIAL FILE	P
4	VIC 20 MERGE	P	9	PRINTER TEST	P	4	CHECK DISK	P	13	RANDOM FILE	P
1	D.AC MERGE	P	4	DISK ADDR CHANGE	P	14	DISPLAY T&S	P	558	PROSTIH BLOKOV	*

DISK. COMMODORE 64 ID=64 15.10.84

27	FROGGER	P	7	N3 OTKIE	P	6	N5 GONG	P	26	BLACKJACK	P	2	S3 DEMO 3	P
40	N1 [illegible]	P	17	EMULATOR	P	2	N6 BOMBE	P	6	S1 FARBDEMO	P	2	DEMO BOOT	P
9	N2 SYNTHESIZER	P	3	GRAFIK	P	2	N7 SCHUSS	P	4	S2 HIRESDEMO	P	0	CCC1	P
71	SPRITES	P	1	DOS BOOT	P	2	N8 SOUND	P	3	S3 GRAFIK	P	17	DEMO C000	P
3	MUSIC	P	4	DOS 5.1	P	37	TABGARTEN	P	2	S4 FARBDEMO	P	82	DEMO?	P
6	SPRITE.MON	P	3	CHANGE 1541	P	47	BITS UND BYTES	P	2	S5 HIRES LOADER	P	10	AKTIVFILTER	P
7	POLLY	P	7	PIANO	P	5	BYTSPRITES	P	32	HIRES DEMO	P	0	PROSTIH BLOKOV	*
7	[illegible]	P	6	N4 YANKEE	P	5	DTSCH TASTATUR	P	4	S1 DEMO 1	P			
6	[illegible]	P	1	FARBTEST	P	30	KALENDER	P	4	S2 DEMO 2	P			

MALI OGLASI

ZA COMMODORE 64 RAZPRODAJA PROGRAMOV PO NAJNIŽJIN CENAH Cene programe od 30.00 do 60.00 dinarjev. Zahtevajte najnovejši spisek 650 programov. Informacije po tel. (061) 373-138 TX 220

BRITANIA SOFTWARE Velika izbira najnovejših programov za ZX spectrum. Programe za katere še niste slišali lahko naročite pri nas po izredno nizkih cenah. Na voljo je tudi več paketov z najnovejšimi igrami. Naša lestvica: Sherlock Holmes, Combat Lynx, White Lighting, Oasis, Tornado Low Level, Full Throttle, Zipper Flipper. Za obširan katalog pišite na naslov: Orikan Horvat, Pod Gradiščem 1/a, 62000 Maribor ali informacije po tel. (062) 24-721 TX 222

GENERATION SOFTWARE! Tudi le mesec z dopolnjenim katalogom programov za spectrum. Najnovejši, izbrano najkvalitetnejši programi: Sherlock Holmes, Full Throttle, Hulk, Android 2, Lords of Midnight, Hobbit 2, T. L. L., Disco Day, Zipper Flipper, White Lightning ter še mnogi drugi lehko postanejo tudi vaši. Velika izbira tudi uporabnih in sistemskih programov ter pakelne ponudbe s popusti. Za brezplačen katalog pišite ali kličite: Dejan Musko, Prešernova 12, 62000 Maribor, telefon (062) 28-547 med 15. in 16. uro TX 221

COMMODORJEVCI, POZOR! Popravljam Commodorjeve kasetofone in po njihovem načrtu izdelujem vezje za snemanje programov na navaden kasetar. Tel. (061) 347-223 Dušan TX 1023

NAJBOLJŠA PONUDBA! Najnovejši programi za spectrum samo 50 in 70 din. Brezplačen spisek. Skripta menjava. Dragan Shapdinović, Gundulićeva 12, 34300 Aranđelovac, tel. (034) 714-946 TX 214

COMMODORE 64 500 programov. Opis vsakega programa, literatura, brezplačen katalog. Popnašil OO software, Skočilovići 9, 41000 Zagreb TX 206

PROGRAME za commodore 64 prodam. Đukić, Zagreb, Čalogovićeva 5, telefon (041) 686-004 TX 202

ZA SPECTRUM super poceni programi. Izberite 15 iger za samo 1000 d.n. Komplet 320 programov samo 6000 din. Velike izbire 370 programov, velike izbire knjig, zaželena izmenjava. Zamenjujem programe za commodore 64. Milivojević Predrag, General Ždanova 30, Beograd, telefon (011) 347 957 TX 203

DANTON STUDIO najnovejši angleški programi za spectrum, navodila, tornado, SchErlok Holmes, tenis, avalon, combat lynx prodam. Cene ugodne, katalog brezplačen. Telefon (071) 514-777 Tašul Daniel, Slaka Skenderove 3, Sarajevo TX 204

VELIKA IZBIRA programov za ZX spectrum. Brezplačen katalog. Ugodne cene. Večja naročila – popust. Zanesljive vrailikacija in hitra dostava. Sonnenschein David, Mlinska pot 17, 51231 Črnuče, telefon (061) 314-918 TX 206

NOV TI 99/4A s kasetofonom in programi prodam. Aleš Krušič, telefon (065) 31-430, Sijenkova 41, 65290 Šempeter pri Gorici TX 184

COMMODORE 64 programe prodam in menjam. Kacin, Begunjska 15, Kranj TX 215

ZX SPECTRUM in **COMMODORE 64 SOFTWARE** – programi in literatura za najpopularnejše mikroračunalnike. Video igre, svanimse jeziki in sistemski programi. ZX SPECTRUM: Lords of midnight, tennis, T. L. L., Combat Lynx, BETA BASIC, Micro Prolog, Pascal, asembleji. Literatura: strojni jezik, ROM disassembled. COMMODORE 64: Chinese juggler, pool, billiard, soccer II, Simmons BASIC 2 (vgrajen FAST + monitor), Pascal 64. Literatura: C-64 tips and tricks, prevod navodil. Pišite za spisek. Naslov: Meljaž Vogel, Trg 44, 62331 Prevalje TX 217

PREVOD NAVODIL za commodore 64 (700 din) prodam. Telefon (061) 559 466 541

SPECTRUM silwersoft izbira novih in starih programov: FULL BIP BOP OF ARTU, SUPERFLIPPER, JUNGLER, SABRE WULF, nizke cene, brezplačen katalog. Telefon (061) 453-952 Marjan TX 209

SPECTRUM hardware avtomatska izbira telefonske številke le sten menik. Informacije po telefonu (061) 575-263 Mojca 543

IGRALNO TASTATURTO za vse spectrumove igre (šest logično razporejenih kurzorjev z vmesnikom – možnost priključitve igralne palice) ugodno prodam. Bertoncelj, Britof 16, 64000 Kranj TX 207

ZA SPECTRUM 16/64 kb ZX 81 1/16 kb najpopolnejši slovenski FRT prevod navodil in basic programiranje, najnovejše programe na kasetah, listingih – top lestvica: MATCHPOINT, MANIC MINER, III FULL THROTLLE, PINSALLWIZZARO, VII SABRE WULF, VII WORLDCUP FOOTBALL. Brezplačen katalog! Telefon (061) 447 156 B 452

PROGRAMI za spectrum (29–39–49 din) brezplačen katalog. Stan naročnik 10% popust. Prodam profesionalno lastaluro za spectrum – »špice« (brez ohišja) za 7500 din. Naročila na naslov: Špenović Borut, Cesta v Meatni log 70, Ljubljana 544

2-BITNI AUT PORT – silwersoftni preklopnik za 80 K spectrum (1500 din), program za mobo printer GP 100 VC na ZX 81 in spectrum prodam. Tel. (069) 23-788 TX 216

SPECTRUM NOUSE – specialist za ZX. Pri nas boste dobili najboljše! Hit programi z angleških lestvic: Top Ten, Top Twenty, Utility Listingi, Literatura. Milen Vrca, Zenje Vujoševića 78, 11070 Novi Beograd TX 211

KORAK NAPREJ S SPECTRUMOM! Programi: statistika, varianse, regresije, linprog, tranprog, annuity, utility, player (PRINT v srbohrvaščini ali navodilo). Literatura v angleščini in nemščini, listingi. Tudi ponudba uporabniških programov in iger za commodore 64. NEW DATA, D. Brašovana 8/10, 21000 Novi Sad TX 212

PRODAJAMO profesionalno tipkovnico za ZX 81 in ZX spectrum ter usmernik za galaksijo. Za vse informacije pokličite tel. (053) 71-005, Elektronika Buje, V. Nazora 3, 51460 Buje TX 213

MINICOM SOFWARE ZA SPECTRUM 48 K

– SSTV (Slow Scan Television) sprejemanje in oddajanje brez hardware 2500 din

– SSTV monitor (samo sprejemanje – hardware ni potreben) 1450 din

– MORSE coder/decoder (20 1500 zn/min) sprejemanje, oddajanje, avtomatsko nastavljanje hitrosti 1450 din

MINICOM PRINT (42 znakov x 29 vrstic, nove funkcije AT POINT, TAB POINT) 900 din

* POPUST – vsi programi skupaj 5500 din

Cena vključuje kaselo C 12, navodilo in poštnino. Dostava po povzetju. Naslov: Roman Laviiv, V. Kecmanovića 27/II, tel. 32 339 ali Ninoslav Jankijević, I. Andrića 47, tel. 32 348, 78000 Banja Luka TX 210

PRODAM SPECTRUM 48 K s programi. Tel. (061) 552 156 TX 218

RAZŠIRJAM spectrum iz 16 K na 48 K. Delo stane 900 din. Serviseram okvare na spectrumu. Vgrajujem reset tipko za 220 din in prvi reset za 190 din, napetostni stabilizator za 790 din ter profesionalno izvedbo za 390 din. Inštruiram basic za 220 din. Tel. (061) 612-546 zvečer 545

PRODAJAM AKCIJSKE IN LOGIČNE igre za računalnik galaksija. Zahtevajte brezplačen katalog. Miloš Smole, Ulica mladinskih brigad 10, 64000 Kranj TX 1021 St

ZVEZA ORGANIZACIJ ZA TEHNICNO KULTURO SLOVENIJE

VABI

k udeležbi na **DRUGEM MIKRORAČUNALNIŠKEM SEJMU** od 21. do 23. DECEMBRA 1984 V CANKARJEVEM DOMU

V PETEK 21. DECEMBRA BO POSVETOVANJE **»RAČUNALNIŠKO IZVENŠOLSKO IN PROSTOČASNO IZOBRAŽEVANJE«** (mladine in odraslih)

Vsi, ki bi radi sodelovali na sejmu oziroma posvetovanju, naj se prijavijo na naslov:

ZVEZA ORGANIZACIJ ZA TEHNIČNO KULTURO SLOVENIJE
LEPI POT 6, p. p. 99, 61001 LJUBLJANA

in navedejo želeno obliko sodelovanja.

O podrobnostih bomo obvestili na naslove prijavljenih.

Šola programiranja v strojnem jeziku (5)

ŽIGA TURK

V prejšnjih nadaljevanjih sem poskušal naštati in razložiti vse osnovna mnemonika, ki sestavljajo kodo Z 80. V nadaljevanju bomo obdelali še ukaze, ki se nanašajo na neposredan stik procesorja z zunanjim svetom IN, OUT in INTERRUPT

Pisanje programa v strojnem jeziku pomeni pisati program stroju »na kožo«. Ne samo procesorju, ampak vsemu računalniku. Poznavanje okolja, v katerem bodo programi delovali, je za dobre programe bistvenega pomena. In nasprotno. Poznavanje strojnega jezika, še bolj pa delovanja računalnika, lahko prihrani marsikatero rutino v strojnem jeziku.

V kodi Z 80 je na voljo 8 parov registrov. To je v primerjavi z drugimi osembitnimi procesorji mnogo, na vsak način pa premalo, da bi računalnik lahko vse podatke hranil v njih. Tudi programi v ROM si morajo sem in tja kaj zabeležiti. Po ukazu DRAW si npr. kaže zadnjo točko zapomniti kje v RAM in ne zasesti registra, ki bo čakal, da bomo spet kaj narisali. Kot beležnica za to, da sprostimo kakšen register, nam lahko rabi sklad (PUSH in POP). Za trajnejše shranjevanje podatkov pa ni primeren, saj se začne dolo z njim zapletati takoj, ko se podatki pomešajo z naslovi »RETURN«.

V programih, ki jih bomo pisali, si bomo rezervirali nekaj bytov, ki nam bodo rabili kot dodatne spremenljivke. Zaradi preglednosti se dogovorimo, da zloge, katerih začetna vrednost ima pomen, definiramo z »DEFB« ali »DEFV«, za tiste, ki jih bomo definirali v programu, spustimo prostor z DEFS. Če imamo na voljo makroasembler, npr. DEVPAC 3.2 ali DEVPAC 4, si lahko delo na račun spremenljivk v pomnilniku močno poenostavimo.

Sistemske spremenljivke v ROM (naštete so na straneh 173–176 priročnika za uporabo računalnika ZX spectrum) so »spominčki«, kamor rutine spravljajo podatke in jih ob ponovnem klicu te rutine spet potrebujejo. Prav zato nam sistemske spremenljivke dajejo vpogled v delo sistema. Če jih spremenjamo, pa lahko sistemske rutine tudi prelisičimo. Veliko je bilo napisano, kako računalnik »sesuti« s pokanjem spremenljivk, manj je bilo povedano, zakaj se to zgodi. Koristni poki so se skoraj vsi nanašali na piskanje tipkovnice in neomejeno pomikanje. Dobro poznavanje, kako delujejo sistemski podprogrami v ROM, nam lahko pomaga, da tudi kakšno na videz nenevarno spremenljivko koristno uporabimo.

Kdor je kdaj pisal programe z velikim številom podatkov, se je gotovo srečal s problemom, kako jih shraniti na trak. Vzemimo, da smo rezultate shranili v obliki numeričnega polja, matrike a. Spraviti na trak je ni težko, preprosto uporabimo SAVE »ime« DATA a() in z LOAD jo znamo tudi kaj hitro naložiti nazaj. Problem pa nastane, ko je treba npr. kakšni zanki, ki elemente izpisuje, povedati, kako veliko je to polje. Hackerji vedo, da je v ROM podprogram, ki poišče naslov spremenljivke. Toda to še ne pomeni, da ga ne moremo poiskati kar iz basica. Pomagala nam bo ena od tistih na videz nezanimivih, s črko »N« označenih sistemskih spremenljivk, ki prav zato, ker njeno pokanje ne »sesuje« programa, nikogar kaj posebno ne privlači. S programom »i« bomo poiskali velikost polja.

Še ena sistemska spremenljivka, označena z »N«, je lahko zelo koristna: DEFADD na naslovu 23563. Redki so mojstri, ki ves program zapišejo v čisti strojni kodi. Večina se zadovolji z rutino ali dvema, ki pospešita izvajanje tistih najzamudnejših operacij. Kot že veste, podprogram v strojni kodi kliče funkcija USR, ki pa razen naslova ne jemlje parametrov. Strojno rutino torej lahko pokličemo, ne moremo pa ji posredovati podatkov, npr. prve in zadnje vrstice v rutini za brisanje večjega števila vrstic naenkrat. Navadno to počnemo z uporabo »javk«, fiksnih pomnilniških lokacij, ki jih popokamo iz basica in njihovo vsebino v strojnem programu preberemo. Klicu podprograma je zato sledil daljši ali krajši »pokajoč« stavek, ki je bil posebej zanimiv, kadar je bilo treba prenašati števila, večja od 255, da o znakih in realnih spremenljivkah niti ne govorimo. Poleg tega so bile stalno težave, kako izbrati te pomnilniške lokacije, da ne bi bilo treba ob vsaki spremembi strojnega programa popravljati klicev v basicu, in po Murphyjevem zakonu se zgodi še to: »Dve rutini v istem programu zagotovo uporabljata isto lokacijo.«

Tudi sam sem dolgo klical podprograme na ta način. Nekoč pa mi je pogled po naključju ostal na spremenljivki DEFADD, tu naj bi se hranil naslov argumenta uporabniško definirane funkcije (user defined function). Klicanje programov v strojnem jeziku postane s tem pravi užitek, še več, z definiranimi funkcijami lahko zelo preprosto in brez poznavanja trikov, ki so bili opisani v prvi in tretji številki MM, širimo basic. Ob klicu funkcije se namreč vrednosti parametrov izračunajo v skrita mesta v program, podobno kot so v programu zapisana števila.

Na ime prvega parametra kaže DEFADD. Podprogram v strojnem jeziku mora torej samo prebrati podatke s teh lokacij. Ker bomo navadno uporabljali cela števila, bodo njihove vrednosti zapisane na lokacijah (DEFADD)+4,12,20 ... kot je razvidno iz skice.

Za primer sta zapisani funkciji M (od kod kam, koliko), ki premakne »koliko bytov« »od kod« »kam« (rutino že poznate iz prejšnjih nadaljevanj), in funkcija x(a, b), ki bo med obema naredila »xor«. Edino težavo je nekaj časa delal klic podprograma, ki je premaknil stavek, s FN. S klicem USR je program brezhibno deloval. Naloga je zanimiva. HACKERJI, NA DELO!

Sistemske spremenljivke so samo vrh ledene gore. Vsak program, tudi najbolj napeta akcijska igra, semtertja uporablja rutine, ki so zapisane v ROM.

Kot smo povedali že v prvem nadaljevanju, so v ROM shranjeni programi, ki se izvajajo ves čas dela z basicom. Še enkrat naj ponovim, da se vrstice programa v basicu nikoli ne izvajajo, ampak so le navodila programom v ROM.

Poznavanje ROM je za resnega programerja nujno. V tej šoli se bomo naučili klicati nekaj najuporabnejših rutin. Tistim, ki postajajo hackerji z dušo in telesom ter željo za vsako ceno prelisičiti basic, pa ponovno svetujem delo, ki je res zgled, kako razložiti delovanje računalnika: The Complete Spectrum ROM Dissassembly avtorja dr. Iana Logana. Po odhodu Richarda Altwasserja in Stevena Vickersa, očetov roma 16K, je bil dr. Logan tisti, ki je priskočil na pomoč pri razvoju roma za vmesnik interface 1. Knjigo je lani izdala založba Melbourne House Publishers.

Vse programe v ROM lahko razdelimo takole: vhodno-izhodne rutine, urejevalnik basica, interpreter basica, obravnavanje izrazov.

Vhodno-izhodne rutine

Te povezujejo procesor in RAM s strojno opremo, ki ju obkroža. Načelno bi lahko samo z zamenjavo teh rutin prenesli basic ZX v katerikoli računalnik Z 80, vendar delitev v ROM le ni tako stroga. Sem spadajo rutine za branje tipkovnice, krmiljenje zvočnika, snemanje na kasetofon, izpisovanje in branje znakov po različnih tokovih, brisanje zaslona in v bistvu tudi vsi podprogrami za grafiko. Uporaba teh rutin, predvsem kanalov in tokov, zagotavlja, da bo naša programskaa oprema združljiva z drugimi programi in drugo periferno opremo. Če bi npr. program LLIST 642 s kasete RŠ izpisoval na tiskalnik

Pomnilnik ob klicu funkcije:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	....
$\underset{\text{DEFADD}}{\text{'A'}}$	14	0	s	L	H	0	','	8	14	0	s	L	H	0	','	....

L=nizki zlog
H=visoki zlog
s=predznak' (ali 0)
14=oznacuje stevilko

Kako dolocimo velikost enodimenzionalnega polja n:

0	1	2	3	4	5	6	7	8	9	10
"n"	len		1	DIM		elementi				

DEST (→ 6)

```
10 LET A=n(1)
20 LET n(1)=PEEK 23629+256*PEEK 23630
30 PRINT "DIM=";PEEK(N(1)-1)+256*PEEK(N(1))
```

skozi #3 in ne s COPY, bi ga bilo mogoče brez vsake spremembe uporabiti tudi za izpis ne drug tiskalnik. Z uporabo zakonitosti kanalov je mogoče napisati en sam program, ki bo iz vsakršnega izhoda kateregakoli programa, ki spoštuje te zakonitosti, naredil datoteko poljubne oblike.

Urejevalnik

Programi, ki sestavljejo urejevalnik programa v basicu, so tesno povezani s samim interpreterjem. Kot ste opazili, ne morete zapisati napačne vrstice, saj se sintaksa sproti kontrolira. To delajo iste rutine, ki pozneje program izvajejo, le da nekje v sredini zavijejo na drugo pot. To morda res zagotavlja enakost sintakse v urejevalniku in med potekom programa, vpliva pa še na nižjo hitrost izvajanja. Tudi rutini, ki ju uporablja program za brisanje, uporabljata tako urejevalnik kot interpret med izvajanjem programa. Rutine, ki sestavljajo urejevalnik, so izredno uporabne pri programiranju t. i. programov toolkit.

Interpreter

To je suženj, ki ve, da je »P-R-I-N-T« ukaz z določenimi parametri in ki v končni fazi zahteve klic vhodno/izhodne rutine, ki bo poslala znake v zunanjo enoto. V bistvu gre za zanko, ki v delu pomnilnika, ki pomeni program (ali vrstico EDIT), bere znak za znakom in se po tabelah odloča, kaj storiti.

Zadnji del rome je interpreter za poseben jezik, ki je vdelan za računanje s plavajočo vejico in polji. Podoben je jezikom, ki jih srečamo v programabilnih kalkulatorjih. Operacije opravlja s števili v skladu kalkulatorja (calculator stack).

Nadaljevanje prihodnjič

```
                 10 ;PRENASANJE FARAMETROV S FUNKCIJO
                 20 ;
FDE8             30        ORG  65000
                 40 ;========================
5C0B             50 DEFADD EQU  23563
196E             60 L_ADR  EQU  #196E     ;RUTINA V ROM POISCE NASLO
V VRSTICE
19E5             70 RCLAIM EQU  #19E5     ;RUTINA ZA BRISANJE
                 80 ;========================
FDE8 C316FE      90        JP   X_OR      ;DEF FN X(A,B)=USR 65000
FDEB            100 DLINE  EQU  $         ;DEF FN D(A,B)=USR 65003
FDEB DD2A0B5C   110        LD   IX,(DEFADD)
FDEF CD02FE     120        CALL GET_P1
FDF2 CD6E19     130        CALL L_ADR
FDF5 E5         140        PUSH HL
FDF6 CD09FE     150        CALL GET_P2
FDF9 CD6E19     160        CALL L_ADR
FDFC D1         170        POP  DE
FDFD CDE519     180        CALL RCLAIM
FE00 180E       190        JR   RETURN
FE02 DD6E04     200 GET_P1 LD   L,(IX+4)
FE05 DD6605     210        LD   H,(IX+5)
FE08 C9         220        RET
FE09 DD6E0C     230 GET_P2 LD   L,(IX+12)
FE0C DD660D     240        LD   H,(IX+13)
FE0F C9         250        RET
FE10 D9         260 RETURN EXX
FE11 215827     270        LD   HL,#2758
FE14 D9         280        EXX
FE15 C9         290        RET
FE16 DD2A0B5C   300 X_OR   LD   IX,(DEFADD)
FE1A CD02FE     310        CALL GET_P1
FE1D E5         320        PUSH HL
FE1E CD09FE     330        CALL GET_P2
FE21 D1         340        POP  DE
FE22 7C         350        LD   A,H
FE23 AA         360        XOR  D
FE24 47         370        LD   B,A
FE25 7D         380        LD   A,L
FE26 AB         390        XOR  E
FE27 4F         400        LD   C,A
FE28 18E6       410        JR   RETURN
Pass 2 errors: 00

Table used:   113  from   189
```

Macro assembler

Kako težko programirati v strojnem jeziku (boljši izraz bi bil »pisati programe, ki se izvajajo v strojnem jeziku«, saj programiramo v zbirniku), je odvisno tudi od tega, kako dobro poznate svoj zbirnik. O osnovnih ukazih zbirnikov je bilo že nekaj povedano. Firmi Dasia in Hisoft sta predstavili svoje nova dosežke, ki poleg vseh običajnih dobrot omogočata uporabo makro definicij.

V uvodu k tečaju strojnega programiranja smo zvedeli, da je zbirnik jezik, kjer se en ukaz simboličnega jezika prevede v [illegible] makro zbirnikih pa to ni več čisto res. Iz znanih ukazov si lahko definiramo poljubno število novih ukazov, podobno kot v strukturiranih jezikih. Toda način prevajanja je drugačen. V pascalu npr. se bo novo definirani ukaz prevedel v KLIC na mesto, kjer je procedura definirana. V makro zbirniku pa je definicija le napotek prevajalniku. Na mesto, kjer uporabimo novi ukaz, bo zbirnik prevedel celotno definicijo. Na prostoru torej ne bomo pridobili, le naš izpis bo preglednejši in programiranje lažje.

Recimo da v programu pogosto selimo vsebino pomnilnika z LDIR. Kot vemo, moremo najprej definirati registrske pare BC, DE in HL, nato pa LDIR pomakne BC zlogov iz HL na DE. Makro definicija je videti takole:

PREMAKNI MAC
 ;»PREMAKNI« je makro ukaz
LD BC,=0
LD DE,=1
LD HL,=2
LDIR
ENDM [illegible] števila, pred katerimi smo napisali =, so parametri makro instrukcij. To so seveda lahko samo konstante (labele).

Ko bo pri prevajanju zbirnik npr. naletel na ukaz
PREMAKNI 1024, 1024*16,0
bo na to mesto prevedel kodo, kot da bi napisali tri LD in LDIR ukaz z ustreznimi parametri. V Hisoftovem zbirniku ne smemo uporabljati makro ukazov v makro definiciji, torej gnezdenje ni dovoljeno. Pri uporabi pa moti predvsem to, da je treba prostor makro definicijam določiti s posebnim ukazom, ki izvorno datoteko v pomnilniku pobriše. O velikosti prostora, namenjenega definicijam makro ukazov med prevajanjem, se moramo odločiti že pred začetkom pisanja, sicer sledijo dolgotrajno shranjevanje na trak, rezerviranje prostora in potem ponovno nalaganje. Še sreča, da deluje tudi z mikrotračniki. Boljša rešitev od navedene bi bila, če bi bile makro definicije med prevajanjem spravljene na istem prostoru, kjer se med prevajanjem shranjujejo vrednosti label.

Ekran, pobrisan z desetimi zlogi:

»LDIR je sicer lepa rešitev,« piše Andrej Vilfan iz Ljubljane, »a ne dovolj kratka.« Njegova rešitev je naslednja:

```
      LD   A,58H      2
      LD   HL,4000H   3
zanka LD(HL),O        2
      INC  HL         1
      CP   H          1
      JR   NZ,zanka   2
      RET             1
```

Dušanu Ulbinu iz Maribora je vseeno, če pobriše še par atributov, s čimer je pridobil 1 zlog, a rešitev ni pravilna:

```
      LD H,63         2
      LD A,66         2
      LD (HL),0       2
      INC HL          1
      [illegible]     1
      RET Z           1
      JR LOOP         2
```

Lovec na nagrade Iztok Saje pa je še zlog krajši. Upošteval je, da lahko ROM briše brez škode.

```
Ld HL,1024*22-1
LOOP XOR A
LD (HL),A
DEC HL
OR H
RET Z
JR LOOP
```

Sam ne vidim krajše različice, razen seveda trivialnih rešitev, klicev v ROM. Nagrada leži še en mesec. Če ne bo nihče krajši, dobi kaseto s programi Iztok.

Zastavlje pa novo poučno nalogo: Pobrisati RAM, kateremu [illegible] program na lokaciji 0. Rešitev baje leži v skladu.

C (2)

ANDREJ VITEK

Pri predstavljanju C-ja danes nadaljujemo in končujemo pri njegovih podatkovnih strukturah. Drugi del članka je namenjen Hisoftovi izvedbi tega jezika. Osnovni podatkovni tipi C-ja so kot v večini programskih jezikov znaki (char), cela števila (int), realna števila (float) in realna števila z dvojno natančnostjo (double). Celih števil je več vrst, med seboj pa jih ločimo s pridevniki pred imenom tipa: long int, ki označuje večja cela števila, short int so majhna cela števila, unsigned int pa določa cela števila brez predznaka. Z njimi računamo po modulu 2^n, kjer je n število bitov v celem številu. Pridevnike lahko med seboj kombiniramo, na primer unsigned short int. Seveda se lahko zgodi, da posamezna izvedba med različnimi vrstami števil ne loči.

Pri zapisu celoštevilskih in realnih konstant veljajo običajna pravila, kot jih poznamo npr. iz basica: 1000, 1e3 itd. Vse realne konstante so dvojno natančne, zato zanje poseben zapis ni potreben. Velika celoštevilska konstante označuje črka L za konstanto, sama po sebi pa so taka vse konstante, ki po velikosti presegajo obseg običajnih števil. Enostavno zapišemo tudi osmiške konstante: začnemo jih z ničlo. Pri šestnajstiških konstantah ravnamo podobno, le za ničlo zapišemo še x. Tako je npr. 037 osmiški, 0x1f pa šestnajstiški zapis desetiške 31.

Z znakom med apostrofoma (enojnima narekovajema) zapišemo znakovno konstanto, npr. '0', ki pomeni znak (ne število in ne niz) 0. Posebne znake, ki jih ni mogoče natisniti, zapišemo z njihovo kodo za ubežnim znakom: znak za konec vrstice je npr. '\010'. Nekatere pogostejše nenatisne znake označujejo posebni ubežni pari: znak za novo vrstico '\n'. Z '\0' zapišemo znak s kodo 0, ki po C-jevem dogovoru zaključuje nize. Znakovne konstante so v spominu predstavljene kot kode znakov, ki jih pomenijo. Z njimi lahko računamo in jih zelo svobodno prirejamo, če je treba, tudi celoštevilskim spremenljivkam. Niz vklenemo med (dvojne) narekovaje, npr. »Moj mikro«. V programu lahko nize povsod uporabljamo kot polja znakov, zavedati se moramo le, da za zadnjim znakom niza prevajalnik doda znak s kodo 0. Zato se 'x' in »x« seveda razlikujeta: slednji pomeni niz dveh znakov, x-a in znaka '\0'. Ker lahko med seboj primerjamo le enostavne vrednosti, je primerjava x==»Bingo« napačna.

C pozna več vrst spremenljivk: avtomatične, statične, zunanje in registre. Avtomatične spremenljivke obstajajo le znotraj posamezne funkcije med njenim izvajanjem. Ob klicu funkcije jim C dodeli prostor v skladu, ob vrnitvi iz nje pa dodeljeni prostor sprosti. Argumenti funkcije so tudi avtomatične spremenljivke, ki jim klic funkcije določi ustrezno vrednost. Statične spremenljivke prav tako pozna le posamezna funkcija, vendar se od klica do klica ohranjajo. Zunanje spremenljivke so skupne več funkcijam in lahko rabijo za enostaven prenos podatkov med funkcijami. Registri po značaju ustrezajo avtomatičnim spremenljivkam, če pa je mogoče, jim prevajalnik dodeli prostor v strojnih registrih. Vse spremenljivke programa moramo deklarirati. Deklaracije se pišejo takole:

```
int num,
float PI = 3.1415926,
```

Slednja deklaracija podeli spremenljivki začetno vrednost. Če spremenljivki začetne vrednosti ne podelimo, ima ob začetku nesmiselno vrednost. Če želimo v programu poklicati funkcijo, ki ne vrne celega števila, jo moramo prav tako deklarirati. Na primer:

```
char digit (),
```

Za imena spremenljivk in funkcij veljajo običajna pravila: začeti se morajo s črko, sledijo pa lahko črke, cifre in še nekateri osebni znaki.

Iz naštetih enostavnih tipov lahko v C-ju gradimo nove. Njihov pregled začnimo pri kazalcih. Vrednost kazalca je naslov podatka nekega drugega tipa. Kazalec na celo število deklariramo takole:

```
int *pi,
```

Pri deklaracijah kazalcev je potreben kanec previdnosti:

```
int *fip(),
```

je deklaracija funkcije, ki vrne kazalec na celo število,

```
int (*pti)()
```

pa je kazalec na funkcijo, ki vrne celo število, kar je v C-ju povsem legalna konstrukcija. Do vrednosti, na katero kaže kazalec, nam v programu pomaga unarni operator *, naslov spremenljivke pa dobimo z unarnim &. Tako je vedno

```
*&v==v
```

S kazalci smemo računati: prišteti ali odšteti jim smemo celo število. Vendar ima to računanje posebnost, h kateri se bomo brž vrnili. Za sedaj omenimo le, da veže * šibkeje kot ++, zato vrednost, na katero kaže kazalec p, poveča (*p)++.

Polja so v nejožni zvezi s kazalci. Polje desetih elementov a[0] ... a[9] deklariramo z

```
int a[10],
```

Indeksi torej tečejo od 0 (namesto od 1, kot je to običaj npr. pri basicu). Ta deklaracija rezervira prostor vsem elementom polja, samo pa se obnaša kot kazalec na prvi element polja. Izraz a[i] ima natanko tak pomen kot izraz *(a+i), zato ju lahko vedno uporabljamo kot sinonime. Razlikuje pa se a od kazalca toliko, da ga ne moremo spreminjati: izraz a++ je napačen. Polje je torej neke vrste »konstanten kazalec«.

Tako smo spoznali tudi posebnost pri računanju s kazalci: povečanje kazalca za i poveča njegovo vrednost, to je naslov, za toliko, da kaže za i elementov naprej. Če element zaseda npr. 5 pomnilnih enot, je naslov večji za 5*i. V večini primerov je to prav tisto, kar potrebujemo. Kazalcu lahko priredimo tudi konstantno vrednost, kar pri Mavrici omogoča zelo elegantno pikanje in pokanje pa celo skok v podprogram v romu. Več o tem pozneje.

Še nekje si moramo pogosto pomagati s kazalci – pri funkcijskih argumentih. Ti so, kot vemo, iste vrste kot avtomatične spremenljivke, ki jim klic podeli prave začetne vrednosti. Kakorkoli jim spreminjamo vrednosti, se spremembe ob vrnitvi iz funkcije izgubijo. Če hočemo spremeniti vrednost argumenta, moremo v funkcijo prinesti kazalec nanj. Vrednost, na katero kaže kazalec, lahko vedno spremenimo. Podobno moremo ravnati tudi, ko hočemo kot argument v funkcijo prinesti funkcijo. Prinesti moramo kazalec nanjo in jo poklicati indirektno, npr.:

```
sort (f, l, tab, comp)
int f, l, tab [ ], (* comp) ( ),
{
if (* comp) ( ...
```

Polje je sestavljeno iz več elementov istega tipa. Podatke različnih tipov pa sestavljamo v strukture. Strukturo lahko najprej vpeljemo in poimenujemo:

```
struct datum {
}int dan, mesec, leto;
struct uslužbenec{
char ime 20, priimek 30,
struct datum rojen, sprejet;
long int plača,
},
```

nato pa uporabimo v deklaraciji:

```
struct uslužbenec oddelek H
[50]
```

Do posameznih sestavnih delov nam pomaga pika:

```
oddelek H [26] rojen dan = 31.
```

Vpeljemo lahko tudi kazalce na strukture. Brez tega ne gre pri strukturah, ki se nanašajo nase. Tako strukturo potrebujemo na primer pri seznamu več elementov enakega tipa. Če ne vemo, koliko uslužbencev ima oddelek H, gornjo definicijo strukture dopolnimo s

```
struct uslužbenec {
...
struct uslužbenec * naslednji
},
```

Med podatke o uslužbencu smo tako dodali kazalec, ki pove, kje so podatki o naslednjem uslužbencu. V C je vdelana funkcija alloc, s katero lahko med izvajanjem programa rezerviramo spomin za nov podatek. Tako lahko seznam uslužbencev dopolnjujemo kar med tekom programa: ko dobimo podatke o novem uslužbencu, napravimo v spominu nov podatek tipa struct uslužbenec, ga napolnimo s podatki in ga s primerno preureditvijo kazalcev vrinemo na pravo mesto v seznamu. Na prvega v seznamu kaže poseben kazalec – glava seznama, npr.

```
struct uslužbenec * oddelek H,
```

Do podatkov o uslužbencih oddelka H nam pomaga puščica:

```
oddelek H -> naslednji -> rojen dan = 28
```

kar je isto kot

```
*(* (oddelek H) naslednji) rojen dan = 28
```

Posamezni podatki v strukturi so v spominu nameščeni drug za drugim. Podobna strukturi je unija. Prav tako združuje več vrst podatkov, razlika pa je v namestitvi: posamezni podatki se začno na istem mestu. Za unijo rezervira prevajalnik toliko prostora, kolikor ga je treba za njen najdaljši sestavni del. Tako kot pri strukturi je tudi dostop do elementov unije.

Prirejamo in sploh operiramo lahko le s podatki enostavnih tipov. Polje moramo prepisati z zanko, strukturo pa podatek za podatkom. V pomoč nam je pri tem operator sizeof, ta pove število pomnilnih enot, ki jih zasedajo polje, struktura ali unija. S kazalci je potem prepisovanje enostavno.

Bodi o C-ju samem dovolj. Čisto prijeten jezik je to, čeprav se po čistosti konceptov in sintakse ne more povsem meriti s pascalom. Prav zaradi svojih posebnosti ima morda pred pascalom tudi prednosti, zlasti pri programiranju sistemske programske opreme. Temu pa je konec koncev tudi namenjen.

Hisoftov prevajalnik

Zdaj pa k Hisoftovemu prevajalniku za C. V oceno smo dobili verzijo 1.0, torej prvo javno verzijo. Tako kot pri Sinclairovem QL tudi pri njej iščemo nepeka uporabniki. Poleg tega tej verziji do polnega C-ja še precej manjka. Precej se bodo morali Hisoftovci potruditi tudi, da bo prevajalnik zares rabil svojemu namenu in bo mogoča v C-ju razvijati resno programsko opremo. Kljub vsemu pa je videti, da so si fantje pri Hisoftu z gensom in pascalom nabrali

precej izkušen) Sistem – sestavljajo ga urejevalnik, prevajalnik in run-time knjižnica – je namreč še kar stabilen pa tudi dovolj prijazen za uporabo, zlasti če ste si Mavrico opremili z mikrotračnikom.

Ogled sistema začnimo pri urejevalniku. Precej običajen vrstični urejevalnik je to, povsem v duhu dosedanjih Hisoftovih. Z njim lahko programsko besedilo dodajamo, brišemo, popravljamo, pregledujamo, preberemo ali zapišemo na kaseto ali mikrotračnik. Do posamezne vrstice pridemo z njeno številko. Popravljamo lahko tudi znotraj vrstice. Zlasti v daljših programih je pripravan ukaz za iskanje in nadomeščanje nizov. Tako iz urejevalnika kot iz prevajalnika lahko kaseto uporabimo kot običajno vhodno/izhodno enoto, delo z njo je s programske strani prav tako udobno kot delo z mikrotračnikom. Nerodno je le to, da s C-jem tvorjenih datotek program v basicu ne zna prebrati. POZOR predolgo intenzivno popravljanje brez vmesnega shranjevanja na trak ni zdravo. Zlasti po masivnih brisanjih se sistem rad zruši!

Ko sistem prvič naložimo, se ga shraniti na mikrotračnik. Po nalaganju proti pričakovanju pridemo v prevajalnik in ne urejevalnik. Karkoli natipkamo, se sproti prevede in – na željo – celo izvede. Z ukazom #include zahtevamo prevod programa, pripravljenega z urejevalnikom, če pa za njim navedemo še ime datoteke, prevaja neposredno s traku. To je priročno za večje programe. Prevajanja končamo s simbolom EOT, ki označuje konec podatkov pri branju s tipkovnice in tu je konec veselja: prevedeni program moramo pognati le enkrat, shraniti ga tudi ne moremo. Verjetno bo to eden od prvih popravkov v verziji 1.1. Napake med prevajanjem zna C opremiti tudi s kratko razlago napake, če razlag s posebnim ukazom ne uničimo. Daljši opisi so na razpolago v priročniku.

Prevajalnik je v pomnilniku razdeljen takole: na začetku je (brez basica) prevajalnik, sledijo pa programsko besedilo, prevedeni program in razlage napak. Napake se ob urejanju programa selijo gor in dol. Od konca pomnilnika nazaj pa leže sklad prevajalnikov in programov. Prevajalnika je okrog 25 K. Njegov začetni naslov je 40444 (© I. Saje). Kaže si ga zapomniti, saj se pogosto zgodi, da vas Mavričin sistem vrne v basic. Od tam pa včasih ni vrnitve niti prek gornjega naslova. Pri pregledovanju programa za nalaganje pazite, ne verjemite tistemu, kar vidite!

Programi, ki jih boste napisali, bodo s pridom izkoriščali kar bogat nabor funkcij iz run-time knjižnice. Del te knjižnice je vdelan v prevajalnik, drugi del pa je knjižnica STDIO. LIB na traku s prevajalnikom. Ogledata (in preselita na mikrotračnik) si jo z urejevalnikom. V njej je precej čisto dobrih zgledov za programe v C-ju, največ iz Kernighanove in Ritchijeve knjige. Podrobneje so funkcije komentirane v priročniku.

Preden program prevedete, ga obvezno shranite, sicer lahko ostanete brez njega. Zavedajte se, da omogoča C pisanje zelo nevarnih programov, zato za vse zrušitve ni vedno kriv C. Zelo pogosti vzroki rušenja so napačno število funkcijskih argumentov, napačen tip argumenta ali kaj podobnega.

Na hitro si oglejmo še, kaj Hisoftovemu C-ju manjka do polnega C-ja. Predvsem ne pozna realnih števil. Tudi to bodo verjetno kmalu odpravili, saj jih pascal pozna, za resno delo pa jih bo moral tudi C. Med različnimi vrstami celih števil ni razlik. Škoda, osembitna aritmetika bi prišla presneto prav. Registrskih spremenljivk ni, upoštevajo se kot avtomatične. Prav tako spremenljivkam ne moremo podeliti začetne vrednosti.

Kazalcem ni mogoče neposredno prirediti celega števila. Gre pa posredno takole:

```
typedef char *pchar,
char LASTK,
LASTK=* cast (pchar) 23560
```

S tem smo si ogledali zadnji natipkani znak, ki je shranjen na naslovu 23560 med sistemskimi spremenljivkami. Pikanje in pokanje je tako precej enostavnejše, predvsem pa elegantnejše. S sistemskimi spremenljivkami lahko namreč delamo kot s strukturami. V prevedeni program lahko s stavkom inline vstavljamo strojno kodo. Tako si lahko brez basica privoščimo odpiranje kanalov in podobne zadeve.

Končajmo! Kot jezik je C čisto prijeten, zlasti ko ga vsaj malo obvladamo. Tudi Mavričin C utegne postati prav pripravno orodje za razvoj programske opreme, seveda ko bodo pri Hisoftu odpravili začetne pomanjkljivosti in napake v verziji 1.0: zaprtost, nestabilnost, prehod v basic in nazaj ter nekaj manjših napak. Do tedaj pa ostaje C 1.0 čisto ustrezno sredstvo za učenje. Za primer: kot popoln začetnik sem čez konec tedna sprogramiral skoraj popoln nabor Mavričinih ukazov za risanje. Manjkata le risanje loka (ni realnih števil) in določitev barve roba (ni funkcije out). Programiranje je bistveno prijetnejše kot v basicu, urejanje ravno tako, v hitrosti pa ni primerjave. Tudi delo z mikrotračnikom je prijetnejše kot iz basica, nerodneje je le listati programe ali izpisovati na tiskalnik. Priročnik je dovolj dobra dokumentacija sistema, urejevalnik je enostaven, pa hkrati dovolj močan. Kljub začetniškim pomanjkljivostim daje Hisoftov C čisto prijeten vtis.

Prolog (3)

TATJANA ZRIMEC
MATJAŽ GAMS

V zadnjem nadaljevanju bomo spoznali nekaj primerov za uporabo prologa. Videli bomo, da z malo pisanja (kodiranja) lahko opišemo kar zahtevne probleme in zastavimo veliko zanimivih vprašanj.

Opišimo v prologu relacije zakonca, poročen, otrok. Trditev »Katerikoli osebi A in B sta poročeni med seboj, če obstaja družina, v kateri je A mož in B žena«, ali pa »če obstaja družina, v kateri je B mož in A žena« zapišemo z dvema stavkoma

zakonca (A, B) – družina (A, B, _)

zakonca (A, B) – družina (B, A, _)

Tretja spremenljivka »–« pomeni »poljubno«, da lahko imata otroke ali pa ne. Trditev »Vsaka oseba X je poročena, če obstaja družina, v kateri je X ali mož ali žena« zapišemo s stavkom

poročen (X) – družina (X, _, _), družina (_, X, _)

Znak »;« pomeni v prologu »ali«. Zapisali bomo še trditev »Vsak X je otrok, če obstaja družina, v kateri je X otrok, starša pa sta poljubna«

otrok (X) – družina (_, _, X)

Ko vpišemo še nekaj družin, lahko začnemo postavljati vprašanja.

```
/* Prgram in podatkouna baza relacija
zakonca, Poracen ,otrok  */

zakonca(A, B):- druzina(A, B, _).
zakonca(A, B):- druzina(A, B, _).

Porocen(X):- druzina(X, _, _);
druzina(_, X, _).

otrok(X):- druzina(_, _, X).

/* zapis nekaj  druzin  */

druzina(mira, tom, _).
druzina(beba, Peter, nik).
druzina(toma, ana, bob).
druzina(drasan, olsa, martin).
druzina(maja, bine, blaz).
druzina(taja, tomi, _).
druzina(janez, micka, janezek).
```

Sedaj lahko postavimo nekaj vprašanj

```
yes
| ?- zakanca(beba,X).

X = Pater

yes
| ?- zakanca(drasan,X).

X = olsa
```

```
| ?- Poracen(tom).

yes
| ?- Poracen(martin).
```

Lahko poiščemo vse poročene osebe

```
| ?- Poracen(X).
X = mira ;

X = beba ;

X = tomo ;

X = drasan ;

X = maja ;

X = taja ;

X = janez ;

X = tom ;

X = Peter ;

X = ana ;

X = olsa ;

X = bine ;

X = tomi ;

X = micka ;
```

Poiščimo še nekaj otrok

```
| ?- otrok(kid).
no
| ?- otrok(X).
X = nik ;
X = bob ;
X = martin ;
X = blaz ;
```

Z naslednjim vprašanjem dobimo pare, ki so si v relaciji, zakonca:

```
| ?- zakonca(X,Y).

X = mira,
Y = tom ;

X = bebe,
Y = peter ;

X = tomo,
Y = ana ;

X = drasan,
Y = olga ;

X = maja,
Y = bine ;

X = taja,
Y = tomi ;

X = Janez,
Y = micka ;

X = mira,
Y = tom ,

yes
```

V naslednjem primeru bomo malo detektivi. Z računalnikom bomo skušali odkriti pobeglega voznika pri prometni nezgodi. Imeli bomo dostop do naslednjih podatkovnih baz, zapisanih v prologu

– podatkovna baza vozil, ki je v zavarovalnici in ima obliko vozilo (registrsko št., tip, barva, let– izdelave, ime– lastnika, priimek)

– podatkovna baza voznikov z osebnimi podatki voznik (ime, priimek, spol, višina, teža, barva– las, št– oseb– izkaznice)

– podatkovna baza bivališč iz občine mesta, ki vsebuje bivališče (ime, priimek, ulice, hišna– št, kraj, št– oseb– izkaznice)

– na policiji pa imamo podatkovno bazo oseb, ki so že povzročile prometno nezgodo, prometne– nesreče (ime, priimek, št– oseb– izkaz)

Na sliki vidimo primere teh baz. Zdaj lahko postavimo nekaj vprašanj, prolog pa bo sam poiskal odgovore. Recimo, da hočemo zvedeti, kakšno vozilo ima voznik Tone Jemec. To vprašanje v prologu zapišemo

(vozilo (X, Y, Z, K, tone, jemec)

Zanimajo nas še drugi podatki, npr. ali je lastnik avta tudi voznik in kje živi. Vprašamo

voznik (– – – – – – – –

bivališče(– – – – – –

Vprašanje je lahko tudi bolj zahtevno: poiščimo vsa vozila letnika 1980, znamke fiat–750

vozilo (X, fiat_750, _, 1980, Z, K)

S podpičjem dobimo vse odgovore, ki ustrezajo vprašanju. V prologu je tudi poseben program

```
/* podatkovna baza vozil  */
/* baza ima obliko
vozilo(Registrska_stevilka, Tip, Barva, Let_izdelave, Ime_lastnika, Priimek)

vozilo(lj_235447, zastava_101, rdeca, 1975, peter, nikolic).
vozilo(lj_554788, zastava_750, bela, 1980, ida, frass).
vozilo(lj_465883, renault_4, siva, 1986, brane, ozura).
vozilo(lj_260640, zastava_750, zelena, 1980, tone, jemec).
vozilo(lj_150920, lada, rdeca, 1987, miran, ceklic).
vozilo(lj_354580, zastava_101, rdeca, 1983, kaja, trobec).
vozilo(lj_653845, zastava_101, bela, 1982, tone, jakel).
vozilo(lj_355476, zastava_101, rdeca, 1975, mira, slak).
vozilo(lj_788845, zastava_101, rdeca, 1983, bojan, stanovsek).
vozilo(lj_783845, zastava_101, rdeca, 1984, janez, sreckic).

/* podatkovna baza voznikov */
/* voznik(ime, Priimek, Spol, Visina, Teza, Barva_las, St_osebne_izkaznice). */

voznik(peter, nikolic, moski, 189, 78, plava, st_267507).
voznik(ida, frass, zenska, 170, 55, rjava, st_772902).
voznik(brane, ozura, moski, 173, 80, plava, st_234769).
voznik(tone, jemec, moski, 185, 70, crna, st_347502).
voznik(miran, ceklic, moski, 180, 83, crna, st_830227).
voznik(janez, sreckic, moski, 185, 59, crna, st_552789).
voznik(bojan, stanovsek, moski, 176, 65, plava, st_567886).
voznik(tone, jakel, moski, 175, 105, siva, st_563224).
voznik(mira, slak, zenska, 165, 63, rjava, st_896690).

/* podatkovna baza bivalisc */
/* bivalisce(ime, Priimek, Ulica, Hisna_stevilka, Kraj,,St_osebne_izkaznice),*/

bivalisce(miran, ceklic, trnovska, 20, ljubljana, st_530227).
bivalisce(ida, frass, aremcroiceva, 35, ljubljana, st_772802).
bivalisce(tone, jemec, trzaska, 45, ljubljana, st_347502).
bivalisce(peter, nikolic, pod_kostnji, 33, ljubljana, st_267507).
bivalisce(brane, ozura, trg_revolucije, 34, ljubljana, st_234769).
bivalisce(janez, sreckic, vecna_pot, 55, ljubljana, st_552789).
bivalisce(kaja, trobec, podmiljeva, 26, ljubljana, st_773245).
bivalisce(bojan, stanovsek, mesa_pijade, 176, ljubljana, st_567886).
bivalisce(tone, jakel, trzaska, 75, ljubljana, st_563224).
bivalisce(mira, slak, zasavska, 85, ljubljana, st_896690).

/* podatkovna baza povzrociteljev prometnih nesrec   */

/* prometni_prekrski(ime, priimek, st_osebne_izkaznice). */
prometne_nesrece(peter, mikovec, st_853456).
prometne_nesrece(iada, piber, st_673902).
prometne_nesrece(bozo, krzisnik, st_342220).
prometne_nesrece(tone, jemec, st_347502).
prometne_nesrece(kaja, trobec, st_773245).
prometne_nesrece(peter, vcuk, st_445900).
prometne_nesrece(brane, ozura, st_234765).
```

s katerim dobimo vse odgovore zbrane v seznam (bagof)

Kot vidimo, so lahko pri prologu v podatkovnih bazah vsi argumenti tudi ključi za iskanje. Pri drugih jezikih moramo naprej določiti, kateri podatek v bazi je ključ za iskanje. Pri prologu je to dvignjeno na čisto logično raven, od ključev in podatkovnih baz ni ostalo nič. Uporabnik je samo navedel logične relacije, drugo je naloga prevajalnika in računalnika.

Z zahtevnejšimi vprašanji lahko rešimo bolj zapletene naloge. Vrnimo se k prvotni nalogi, poiščimo pobeglega voznika! Priče so nam dale naslednje podatke: bila je rdeča stoenka in vozil je črnolas moški. Avto je odpeljal tako hitro, da ni bilo mogoče zapisati registrske številke. Vemo še to, da se je nezgoda pripetila v okolici Ljubljane.

Spraševati bomo začeli tako, da bomo postavili nekaj domnev

```
| ?- vozilo(Xst,Xznamka,Xbarva,Xletnica,tone,Jemec).

Xletnica = 1980,
Xst = lj_260640,
Xznamka = zastava_750,
Xbarva = zelena ;.

no

| ?- voznik(tone,jemec,_,_,_,_,_),
|      bivalisce(tone,Jemec,Ulica,His_st,Kraj,_).

His_st = 45,
Ulica = trzaska,
Kraj = ljubljana ;

no

| ?- vozilo(Reg_st,zastava_750,_,1980,Ime,Priimek).

Reg_st = lj_554788,
Ime = ida,
Priimek = Frass ;

Reg_st = lj_260640,
Ime = tone,
Priimek = jemec ;
```

Glede na odgovore nanje bomo postavljali nove domneve in vprašanja, dokler ne pridemo do rešitve.

Domnevamo, da je voznik tudi lastnik avta in da je že povzročil prometno nezgodo.

```
| ?- vozilo(Reg_st,zastava
_101,rdeca,_,Ime,Priimek),
|     prometna_nesreca
(Ime,Priimek,_).

Reg_st = 1J_354880,
Ime = kaja,
Priimek = trobec ;

| ?- vozilo(Reg_st,zastava
_101,_,_,Ime,Priimek),
voznik(Ime,Priimek,_,_,_,_
,_).

Reg_st = 1J_235447,
Ime = peter,
Priimek = nikolic ;
Reg_st = 1J_653545,
Ime = tone,
Priimek = Jakel ;
Reg_st = 1J_358478,
Ime = mira,
Priimek = slak ;
Reg_st = 1J_788845,
Ime = bojan,
Priimek = stanovsek ;
Reg_st = 1J_763545,
Ime = Janez,
Priimek = sreckic ;

| ?- vozilo(_,zastava_101,
rdeca,_,Ime,Priimek),
|     voznik(Ime,Priimek,
moski,_,_,_,_).

Ime = peter,
Priimek = nikolic ;
Ime = bojan,
Priimek = stanovsek ;
Ime = Janez,
Priimek = sreckic ;

| ?- voznik(peter,nikolic,
_,Vis,Tez,Barva_las,_).

Vis = 183,
Barva_las = plava,
Tez = 79

| ?- voznik(Ime,Priimek,
moski,_,_,crna,_).

Ime = tone,
Priimek = jemec ;

Ime = miran,
Priimek = cokiic ;

Ime = Janez,
Priimek = sreckic ;

| ?- vozilo(Reg_st,zastava
_101,rdeca,_,Ime,Priimek),
|     voznik(Ime,Priimek,
moski,_,_,crna,_),
|   bivalisce(Ime,Priimek,
Ulica,Hisna_st,Kraj,St_os).

Hisna_st = 99,
St_os = st_552788,
Reg_st = 1J_783845,
Ulica = vsona_pot,
Ime = Janez,
Priimek = sreckic,
Kraj = Ljubljana ;
```

```
/* podatkovna baza prebivalcev Slovenije    */

:-op(200,xfy,'/').

prebivalec(maja,bicku,st_334568,zenska,samska,plava,65,170,80/70/95).
prebivalec(sonja,ilc,st_387458,zenska,samska,crna,75,180,100/75/95).
prebivalec(teja,miskic,st_132885,zenska,porocena,plava,100,170,80/80/105).
prebivalec(beba,berik,st_233455,zenska,samska,rjava,65,169,95/70/98).
prebivalec(kaja,vinko,st_944002,zenska,samska,plava,58,170,90/68/93).
prebivalec(tamara,lenas,st_472366,zenska,porocena,rjava,65,163,85/65/90).
prebivalec(bojan,stanovsek,st_356986,moski,samski,plava,85,178,100/78/95).
prebivalec(tone,jemec,st_312534,moski,porocen,crna,105,175,100/80/100).
prebivalec(mira,slak,st_134567,zenska,samska,rjava,63,169,89/68/90).
prebivalec(alenka,fink,st_356363,zenska,samska,crna,67,167,85/70/96).
```

Za šalo si zamislimo še nalogo, da moramo izbrati miss Slovenije za leto 1985 med vsemi prebivalci Slovenije. Podatke o prebivalcih moramo dopolniti:

prebivalec (ime, priimek, št–osebne izkaz., spol, stan, barva–las, teža, višina, prsi /pas/boki)

Komisija za izbor najlepše je predpisala naslednja merila in jih vstavila v računalnik:

spol – ženski
višina – med 160 in 170 cm
teža – med 50 in 60 kg
stan – neporočena
mere – pogoji (boki–prsi) <5 cm in (prsi-pas)> 10 cm

Z naslednjim programom bomo izbrali kandidatke, ki so prišle v ožji izbor. (Merili raje ne preskusite na svoji ženi!)

! ? – prebivalec (ime, priimek, – ženska, samska, teža, višina, prsi/ pas/boki), višina = < 170, višina > = 160, teža = < 70, teža > = 60, (boki-prsi) = < 5, (prsi-pas) > = 10

```
Ime = maja,
Priimek = bicku,

Ime = beba,
Priimek = berik,

Ime = mira,
Priimek = slak,

Ime = alenka,
Priimek = fink,
```

Iz dobljenega seznama bo komisija izbrala miss 1985 na prireditvi.

V sestavku smo pokazali nekatere značilnosti prologa, predvsem tiste, zaradi katerih je način razmišljanja pri programiranju v tem jeziku drugačen (bolj naraven) kot v drugih, postopkovnih jezikih. Nekatere naloge lahko rešujemo z izredno malo kodiranja. Že pri programu z desetimi stavki lahko postavimo veliko vprašanj.

Če rešujemo probleme z zbirnikom, ki je najbliže računalniku, zgubimo veliko časa s pisanjem programa, program se pa zelo hitro izvaja in porabi malo pomnilniškega prostora. Pri prologu, ki je bliže človeku kot stroju, je prav nasprotno. Program napišemo zelo hitro, zato pa je počasnejši in porabi več prostora. Kadar programiramo v prologu, prepustimo velik del naloge računalniku (sam poišče in izbere način, kako bo nalogo rešil).

Današnji računalniki sodijo v tretjo generacijo in niso najprimernejši za prolog. Malo bolje bo s četrto generacijo računalnikov, ki je še istega (von Neumannovega) tipa, vendar hitrejša in ima precej več pomnilniškega prostora. Šele peta generacija bo velik skok naprej. Ti računalniki se bodo močno razlikovali od dosedanjih »strojčkov«. S paralelnim izvajanjem, podatkovno vodeno arhitekturo, prologom oz. njegovimi dialekti (lisp), ekspertnimi sistemi, naravnim jezikom (vhod – izhod) in vrhunsko grafiko se bodo močno približali človeškemu mišljenju. Programiranje v strojni kodi ter zbirnih in postopkovnih jezikih bo šlo v staro šaro. Računalniki pete generacije ga ne bodo potrebovali, tako kot danes vozniku ni treba biti mehanik, če se hoče peljati.

Prve črte z računalnikom (4)

ANDREJ VITEK

Danes se z risanjem selimo v prostor, v tri razsežnosti. Ogledali si bomo le nekaj osnovnih principov. Postopki prostorske grafike so namreč bistveno zahtevnejši, tako po čisto programerski strani kot tudi teoretično. V nadaljevanju zato nekaj matematičnega znanja ne bo odveč, zlasti pa bo konstil konec občutka za prostorsko predstavo.

Tudi v prostoru si pri opisu objektov – krivulj, ploskev in teles – pomagamo s koordinatnim sistemom. Koordinatama x in y iz dvorazsežnega sveta se pridružuje še tretja, z. Koordinatni sistem običajno izberemo tako, da označuje koordinata y globino, koordinata z pa višino.

Pri risanju v ravnini nam je to zadostovalo, le včasih smo morali sliko malo povečati ali pomanjšati, da smo jo lahko spravili na zaslon primerno veliko. Pri prostorskem risanju pa moramo pred risanjem poskrbeti še za **projekcijo**, za preslikavo trirazsežnega prostora v dvorazsežno ravnino. Pred isto nalogo so bili slikarji in matematiki že v antični Grčiji, prve razprave, ki so se ohranile do danes, pa so nastale v renesansi. Projekcije so bile dolgo časa gonilo razvoja v geometriji in so še danes eden od njenih temeljev.

Projekcij je več vrst, podrobneje pa si bomo ogledali le dve: vzporedno, ki ji s tujko pravimo aksonometrija, in točkovno, ki ji pravimo tudi perspektiva. Pri obeh projiciramo tako, da skoz izvorno točko pošljemo žarek proti ravnini zaslona. Kjer ta žarek pade na zaslon, nastane slika točke. Žarek je seveda premica. Pri vzporedni projekciji so vsi žarki med seboj vzporedni, pri točkovni projekciji pa se vsi sekajo v izbrani točki prostora, v središču žarkovja. Značilnost teh dveh projekcij je, da preslikujeta ravne črte v ravne črte. Vzporedna projekcija ►

pri tem ohrani vzporednost in velikostne razmerja. V praksi pa so zanimive tudi drugačne projekcije. Na primer tista, ki jo srečamo pri zrcalih na nepreglednih ovinkih ali pri kroglastem zrcalu. Z zanimivimi projekcijami se je – poleg drugega – ukvarjal nizozemski slikar M. C. Escher.

Poglejmo zdaj, kako bi enostavno opisali projekcijo. Ker hočemo z risanjem običajno ponazoriti vtis, ki ga ob pogledu na objekt dobi njegov opazovalec, ravnamo takole. Povemo položaj očišča, to je točko, v kateri je opazovalčevo oko. Poleg nje izberemo drugo točko, gledišče, v katero oko gleda. Premici skoz izbrani točki pravimo os gledanja. Skoz gledišče postavimo pravokotno na os ravnino zaslona. Pri vzporedni projekciji bodo žarki vzporedni osi gledanja, pri točkovni pa bo očišče hkrati središče žarkov. Izbrati moremo še koordinatni sistem v ravnini zaslona. Izhodišče – točko 0,0 – določimo tako, da povemo zaslonski koordinati gledišča (ki je seveda v ravnini zaslona). Koordinati smeri pa izberemo tako, da je smer x v zaslonu v prostoru vodoravna. Pri predstavljanju vam bosta precej pomagala kos kartona (za zaslon) in svinčnik (za os gledanja).

Še nekaj napotkov. Gledišče je pametno postaviti nekam v telo, ki mu v sliki velja osrednja pozornost. Ne glede na tip projekcije, bodo v tem primeru narisane razseznosti približno enake pravim razseznostim telesa. Če pri perspektivi gledišče po osi primikamo očišču, se slika manjša, če ga odmikamo, se slika veča. Perspektivo z enim središčem dobimo, ko je os gledanja pravokotna na prednjo ploskev narisanega telesa, z dvema središčema pa, ko je os gledanja vodoravna.

Zvezo med koordinatami v prostoru in na zaslonu je najlaže poiskati, ko je gledišče v izhodišču prostora (točki 0, 0, 0), oko pa na eni izmed koordinatnih osi, npr. na osi z. Pri vzporedni projekciji se tedaj zaslonska in prostorska x in y ujemata, pri točkovni projekciji pa zvezo izračunamo z uporabo podobnih trikotnikov. Če je višina očišča 1, je $xz=xp/(1-zp/1)$ in podobno za y. Splošni primer, ko sta gledišče in očišče poljubni točki v prostoru, razrešimo tako, da koordinatni sistem pred projekcijo primerno premaknemo in zasukamo. V tem primeru se seveda spremenijo koordinate točk objektov v prostoru. Kako se tako dobljene nove koordinate izražajo s starimi, opisuje devet števil txx, txy, ... tzz, pri čemer je npr.

$$xn=txx*xs+txy*ys+txz*zs$$

Tu je xn nova prostorska koordinata, xs pa stara. Podobni sta izraza tudi za y in za z. Pot do izrazov števil txx ... je precej zamudna in se ji bomo tu odrekli. Vrstica 2100 v programu 16 prikazuje račun teh števil, vrstice od 3100 naprej pa opravijo projekcijo. Omeniti je treba še, da se pri vzporedni projekciji slike različno oddaljenih enako velikih teles ne razlikujejo, pri točkovni pa so bolj oddaljena telesa vse manjša.

Naslednji problem, na katerega naletimo pri prostorskem risanju, je opis objekta, ki ga želimo narisati. Pri krivuljah je opis podoben kot v ravnini, v njem je pač ena funkcija več. Ploskev lahko opišemo kot funkcijo dveh spremenljivk, npr. $z = 1/(1+x^2+y^2)$.

A več o risanju takih ploskev drugič. Telesa pa lahko opišemo na celo vrsto načinov, od katerih je vsak primeren za svojo vrsto uporabe. Za risanje sta najprimernejša dva. Pri prvem telo opišemo z njegovimi oglišči, robovi in stranskimi ploskvami. Pri drugem povemo, kako telo sestavimo iz posameznih enostavnih elementov, npr. kvadrov, valjev in podobnega. Sestavljamo jih lahko z dodajanjem (npr. macola je železen kvader na koncu valjastega ročaja), izrezovanjem (okroglo vrtino iz macole izrežemo z valjem) ali prirezovanjem. Vpeljava ukrivljenih ploskev v sliko prinese precej dodatnih komplikacij. Prva med njimi je že določitev obrisa (silhuete) telesa.

Če se pri risanju zadovoljimo s samo obliko oziroma precej splošnim vtisom o njej, lahko telo predstavimo kar s kletko njegovih robov. V tri polja shranimo koordinate oglišč, v drugi dve pa zaporedne številke začetnega in končnega krajišča robe. Tako je telo opisano v programu 16. Slabe stran takega opisa se kmalu pokaže: poskusite s programom 16 narisati objekt s kakimi 30 robovi! V gosti mreži črt se vsak občutek v obliki povsem zgubi. Pri risanju kompliciranih objektov se moramo zato lotiti odstranjevanja skri-

```
  10 REM    Program 16
  20 REM
  30 REM Risanje telesa
 100 REM Osnovni meni
 110 GO SUB 900
 120 PRINT "0-konec" "1-branje telesa" "2-
     projekcija " 3-risanje" "4-transformacija"
 130 INPUT LINE a$: IF a$<"0" OR a$>"9" THEN
     BEEP 2,24: GO TO 100
 200 IF a$="0" THEN STOP
 210 IF a$="1" THEN GO SUB 1000: GO TO 300
 220 IF a$="2" THEN GO SUB 2000: GO TO 300
 230 IF a$="3" THEN GO SUB 3000: GO TO 300
 240 IF a$="4" THEN GO SUB 4000: GO TO 300
 300 GO TO 100
 900 REM Naslov
 910 CLS : PRINT "1 TRID 1    1984 Andrej
     Vitek "
 920 RETURN
1000 REM Pobatki o osnovnem telesu
1010 LET mobj=10: DIM z(mobj)
1020 LET nobj=1: LET o(1)=1
1100 REM Branje tock
1110 GO SUB 900
1120 PRINT "Koordinate tock '" n ;TAB 4;"x";TAB
     13;"y";TAB 22;"z"''
1130 LET ntck=0: LET mtck=100: DIM x(mtck): DIM
     y(mtck): DIM z(mtck)
1140 INPUT LINE a$: IF a$="" THEN GO TO 1200
1150 INPUT (a$);TAB 10; LINE b$;TAB 20; LINE c$
1160 LET ntck=ntck+1: LET x(ntck)=VAL a$: LET y(
     ntck)=VAL b$: LET z(ntck)=VAL c$
1170 PRINT ntck;TAB 4;x(ntck);TAB 13;y(ntck)
     ;TAB 22;z(ntck)
1180 IF ntck<mtck THEN GO TO 1140
1200 REM Branje robov
1210 PRINT '"Robovi: " n ;TAB 4;"zacetek",TAB
     13;"konec"
1220 LET nrob=0: LET mrob=50: DIM p(mrob): DIM
     q(mrob)
1230 INPUT LINE a$: IF a$="" THEN GO TO 1290
1240 INPUT (a$);TAB 10; LINE b$
1250 LET nrob=nrob+1: LET p(nrob)=VAL a$: LET q(
     nrob)=VAL b$
1260 PRINT nrob;TAB 4;p(nrob);TAB 13;q(nrob)
1270 IF nrob<mrob THEN GO TO 1230
1280 LET ftck=ntck: LET frob=nrob
1290 RETURN
2000 REM Branje projekcije
2010 GO SUB 900
2020 PRINT "Polozaj ocesa"'TAB 4;"x";TAB 13;"y"
     ;TAB 22;"z"
2030 INPUT xoko;TAB 10;yoko;TAB 20;zoko
2040 PRINT TAB 4;xoko;TAB 13;yoko;TAB 22;zoko
2050 PRINT "Polozaj gledisca"'TAB 4;"x";TAB
     13;"y";TAB 22;"z"
2060 INPUT xglb;TAB 10;yglb;TAB 20;zglb
2070 PRINT TAB 4;xglb;TAB 13;yglb;TAB 22;zglb
2080 LET a=xoko-xglb: LET b=yoko-yglb: LET
     c=zoko-zglb
2090 LET s=a*a+b*b: LET l=s+c*c: LET s=SQR s:
     LET l=SQR l
2100 IF s>0 THEN LET txx=-b/s: LET tyx=-a*c/(s*
     l): LET tzx=a/l: LET txy=a/s: LET tyy=-b*c/
     (s*l): LET tzy=b/l: LET txz=0: LET tyz=s/l:
     LET tzz=c/l
2110 IF s=0 THEN LET txx=SGN c: LET tyx=0: LET
     tzx=0: LET txy=0: LET tyy=1: LET tzy=0:
     LET txz=0: LET tyz=0: LET tzz=SGN c
2120 PRINT "Polozaj gledisca na zaslonu"'TAB
     4;"x";TAB 13;"y"
2130 INPUT xzas;TAB 10;yzas
2140 PRINT TAB 4;xzas;TAB 13;yzas
2150 PRINT "Povecava/pomanjsava ";
2160 INPUT pov
2170 PRINT pov
2180 PRINT "0 aksonometrija " 1 perspektiva
2190 INPUT pers
2200 RETURN
3000 REM Risanje
3010 GO SUB 900
3020 LET m=1: LET n=nobj
3030 INPUT "Objekte od "; LINE a$: IF a$=""
     THEN GO TO 3050
3040 LET m=VAL a$: INPUT " do "; LINE a$: LET
     n=m: IF a$<>"" THEN LET n=VAL a$
3050 FOR k=m TO n
3060 FOR r=o(k) TO o(k+1)-1
3070 LET i=p(r): LET j=q(r)
3080 LET a=x(i): LET b=y(i): LET c=z(i): GO SUB
     3500: LET xz=xt: LET yz=yt
3090 LET a=x(j): LET b=y(j): LET c=z(j): GO SUB
     3500: LET xk=xt: LET yk=yt
3100 PLOT xz,yz: DRAW xk-xz,yk-yz
3110 NEXT r
3120 NEXT k
3130 PAUSE 0
3140 RETURN
3500 REM Projeciranje
3510 LET a=a-xglb: LET b=b-yglb: LET c=c-zglb
3520 LET xt=txx*a+txy*b+txz*c
3530 LET yt=tyx*a+tyy*b+tyz*c
3540 LET zt=tzx*a+tzy*b+tzz*c
3550 IF pers THEN LET zt=1-zt/l: LET xt=xt/zt:
     LET yt=yt/zt
3560 LET xt=INT (xzas+pov*xt): LET yt=INT (yzas+
     pov*yt)
3570 RETURN
4000 REM Transformacija
4010 IF nobj>=mobj OR ntck+ftck>mtck OR nrob+
     frob>mrob THEN PRINT "Ni prostora za nov
     objekt ": PAUSE 100: RETURN
4020 GO SUB 900: LET nobj=nobj+1: LET o(nobj)=
     nrob+1: LET ktck=ntck
4030 PRINT "Faktorji povecave"'TAB 4;"a";TAB
     13;"b";TAB 22;"c"
4040 INPUT LINE a$: IF a$="" THEN LET a=1: LET
     b=1: LET c=1: GO TO 4060
4050 LET a=VAL a$: INPUT (a);TAB 10;b;TAB 20;c
4060 PRINT TAB 4;a;TAB 13;b;TAB 22;c
4070 PRINT '"Kot zasuka ";
4080 INPUT LINE a$: LET fi=0: IF a$<>"" THEN
     LET fi=VAL a$
4090 PRINT fi
4100 PRINT "Premik "'TAB 4;"dx";TAB 13;"dy"
     ;TAB 22;"dz"
4110 INPUT LINE a$: IF a$="" THEN LET dx=0:
     LET dy=0: LET dz=0: GO TO 4130
4120 LET dx=VAL a$: INPUT (dx);TAB 10;dy;TAB
     20;dz
4130 PRINT TAB 4;dx;TAB 13;dy;TAB 22;dz
4140 REM Transformacija tock
4150 LET cfi=COS fi: LET sfi=SIN fi
4160 FOR i=1 TO ftck
4170 LET ntck=ntck+1
4180 LET xt=a*x(i): LET yt=b*y(i): LET zt=c*z(i)
4190 LET x(ntck)=dx+cfi*xt-sfi*yt: LET y(ntck)
     =dy+sfi*xt+cfi*yt: LET z(ntck)=dz+zt
4200 NEXT i
4210 REM Novi robovi
4220 FOR i=1 TO frob
4230 LET nrob=nrob+1
4240 LET p(nrob)=ktck+p(i): LET q(nrob)=ktck+q(
     i)
4250 NEXT i
4260 RETURN
```

tih robov in ploskev. V tem primeru moramo poleg robov telesa poznati tudi njegove ploskve, zlasti še če hočemo telo narisati pravilno osenčeno in pobarvano. Več o risanju s skritimi robovi in ploskvami prihodnjič.

Danes se seznanimo še s transformacijami v prostoru, s premikom, povečavo in zasukom. Z njimi si precej poenostavimo sestavljanje okolja. Naselje več enakih hišic sestavimo tako, da hišico, njena oglišča, robove in ploskve podamo le enkrat, nato pa ta osnovni model s postavitvenimi transformacijami razpostavljamo po prostoru. Tako prihranimo pri pomnilniškem prostoru, potrebnem za opis naselja.

Premik podaja trojica števil, ki pove, za koliko premik spremeni posamezno koordinato točke. Premik za dx, dy, dz točko s koordinatami x, y, z preslavi v točko x+dx, y+dy, z+dz.

Tudi **povečavo** podaja trojica števil, ki povedo, za koliko se poveča telo vzdolž posamezne koordinatne smeri. Če so ta števila večja od 1, telo zraste, sicer se skrči. Povečava s faktorji a, b, c namreč preslika koordinate takole: točko x, y, z v točko a* x, b* y, c* z.

So tudi bolj splošne povečave, vendar jih le redko potrebujemo. Še najbolj zamotan pa je opis **zasuka**. Zato se bomo pri njem omejili le na en zelo poseben primer, ko sukamo okrog osi z (navpične premice, na kateri je x=y=0). V tem primeru zasuk podamo z enim samim številom fi – kotom zasuka.

Točko x, y, z preslika zasuk v točko

x* cos(fi)–y* sin(fi), x* sin(fi)+y* cos(fi), z

Višina se ohranja. Pri splošnejših zasukih lahko izberemo za os poljubno premico. Postavitveno transformacijo torej določa sedem števil: faktorji povečave, kot zasuka in spremembe koordinat. Ker ni vseeno, v katerem vrstnem redu opravimo posamezne transformacije, velja dogovor, da telo najprej povečamo, nato zasukamo in nazadnje premaknemo. Vrstice od 4000 naprej v programu 16 iz osnovnega objekta, ki ga podamo, z gornjimi transformacijami dodajajo v seznam objektov nove.

Nadaljevanje prihodnjič

Mikro Assembler

GOJKO JOVANOVIČ

Ameriški strokovnjak za mikroračunalnike Don Lancaster je v eni svojih knjig zapisal: »Popolnoma nemogoče je napisati dober program v basicu. Pascal, se razume, je tako slab, da ni vreden omembe. Dobri programi MORAJO biti v celoti ali delno napisani v strojnem jeziku.« Njegove besede so sicer namenjene lastnikom mikroračunalnika apple, vendar veljajo prav tako za C 64.

Commodore 64 je baje odličen računalnik (vsi prospekti zatrjujejo tako), vprašanje je le, kje se njegove izvrstne lastnosti skrivajo. Vdelani basic je obupen: programiranje kratke melodije zahteva vsaj srednjo glasbeno izobrazbo, do grafike pa pride človek šele potem, ko je spikal in spokal polovico naslovov v pomnilniku. Na prvi pogled torej prav zanikrne naprave. No, vse le ni tako črno. Svojega ljubljenca se moramo lotiti na pravi način in pokazal bo vse, kar zna. Tega ni malo. Pravi način, to pomeni programirati v assemblerju, zbirnem jeziku.

Ključ nam ponuja podjetje Supersoft v rjavi škatlici (malo debelejši od kasete) z imenom MIKRO ASSEMBLER. Škatlico preprosto vtaknemo v zadek svojega C 64 in že imamo na voljo tri programe: monitor, editor in assembler.

Oglejmo si najprej editor. To je običajen zaslonski urejevalnik. Nekatere njegove ukaze lahko uporabljamo tudi v basicu. To velja za ukaz AUTO, ki avtomatično številči programske vrstice z želenim začetkom in korakom. Ukaz DELETE omogoča brisanje posameznih vrstic ali celotnega bloka programa.

Večina ukazov je seveda namenjena programiranju v zbirnem jeziku. Tako ukaz FIND poišče vse vrstice, v katerih je tisto, kar iščemo. FIND LDA bo izpisal vse vrstice, je podoben ukazu LIST, le da nam izpiše program izredno pregledno. Ukaz TABLE izpiše abecedno urejen seznam vseh etiket (label) in njihovih vrednosti, ki smo jih v programu uporabili.

Vsi, ki ljubijo matematiko, bodo zadovoljni z ukazom NUMBER. Z njim pretvarjamo števila v različne številske sestave. NUMBER 24576 bo izpisal:

$6000 24576 e060060
%0110000000000000

$ označuje šestnajstiško, e osmiško, % pa dvojiško število.

Obenem so nam na voljo ukazi, ki jih pozna basic: LIST, SAVE, LOAD itd.

Editor je seveda najbolj skromen program, ki se skriva v škatlici. Odtipkajmo ukaz TIM in znašli se bomo v čudnem svetu številk in črk, v svetu monitorja. Kaj nam monitor ponuja? Pri večini monitorskih ukazov moramo ob črki, ki pomeni ukaz, zapisati tudi to, na kateri naslov oziroma del pomnilnika se nanaša. Vsebino pomnilnika je možno pregledovati z ukazom M (M 0C00 0D000), pri čemer računalnik na zaslonu izpiše naslove in njihovo vsebino. V vsaki vrsti je osem naslovov in vrednosti. Prikazovanje lahko ustavimo s tipko RUN/STOP. Ukaz D (disassemble) nam bo vsebino naslovov prevedel v zbirni jezik in jo izpisal v obliki oštevilčenih vrstic.

Izvajanje programa sprožimo z ukazom G (execute). Ukaza L (load) in S (save) sta nam znana že iz basica, navesti pa moramo tudi obseg pomnilnika, ki naj se naloži ali shrani. Koristen je ukaz H (hunt), ki deluje podobno kot FIND. Ob ukazu moramo navesti, katero vrednost iščemo. Npr. H 0000 FFFF EA bo poiskal in izpisal vse naslove, ki vsebujejo to vrednost (to je koda ukaza NOP). Če želimo pregledati vsebino registrov, zadostuje pritisk na tipko R. Ukaz T (transfer) prenese vsebino naslovov iz enega dela pomnilnika v drugega. Če nas muči domotožje po basicu, pritisnemo X (exit) in zapustili bomo nenavadno monitorsko deželo.

Končno je tu še assembler, zaradi katerega smo pravzaprav dvignili ves ta prah. Mikro assembler je tripasoven, kar pomeni, da trikrat »pregleda« izvorni program, preden ga prevede v strojno kodo. Prednost takega zbirnika je, da lahko uporabljamo etikete, ki jih definiramo šele po njihovi uporabi. Tudi odkrivanje sintaktičnih napak je zelo temeljito. Kljub tripasovnosti je mikro assembler zelo hiter, zlasti če ga primerjamo z interpreterjem za basic. Izvorni program prevajamo z ukazom ASSEMBLE. Pri tem se na zaslonu izpišeta začetni in končni naslov, kjer je prevedeni program shranjen. Seveda lahko tudi sami določimo, kam naj se prevedeni program shrani. Nasproten ukaz je DISASSEMBLE, ki deluje tako kot ukaz D v monitorju.

Omenili smo že, da lahko mikro assembler uporablja etikete. Na voljo pa je še cela vrsta psevdo ukazov (te ukaze računalnik sam prevede v pravilno strojno kodo). Med njimi je najvažnejši ukaz *=, s katerim določimo naslov, kamor naj assembler shrani strojno kodo. *= C000 pomeni, naj se program, ki sledi temu ukazu, shrani od naslova C000 naprej. Ukazi WOR, BYT in TXT so za vnašanje podatkov v pomnilnik. Z WOR vnašamo 16-bitne vrednosti po načelu LOHI (prvi se shrani nižji byte, nato višji). BYT je v bistvu enak ukazu WOR, le da z njim vnašamo 8-bitne vrednosti. Celoten niz znakov (string) lahko shranimo z ukazom TXT (npr. 170 TXT »TO JE NIZ ZNAKOV«). Vrednosti lahko vnašamo v desetiški, dvojiški, osmiški in šestnajstiški obliki ali kot vrednosti ASCII, pri čemer označimo vrednost z opuščajem (npr. LDA # R). Komentarji so lahko dolgi do 80 znakov, uvedemo pa jih s klicajem (npr. 160! TO JE KOMENTAR).

Z mikro assemblerjem lahko pišemo programe, dolge 30 K, strojna koda pa bo obsegala le 3 ali 4 K. Če želimo vnesti daljši program, pozna mikro assembler ukaz LNK, s katerim povežemo posamezne dele programa. Povezovanje končamo z ukazom END. Tako sestavljamo izredno dolge programe, saj lahko strojna koda obsega do 12 K. Lastniki tiskalnikov bodo koristno uporabili ukaz OUT, s katerim izpišemo izvorni in prevedeni program na papir ali zaslon. Zapisovanje prekinemo z ukazom OFF ali END.

Pri ugotavljanju sintaktičnih napak, ki je v strojnem jeziku verjetno najtežje opravilo, nam assembler pomaga tako, da izpiše številko vrstice in vrsto napake, podobno kot v basicu.

Še beseda o pomanjkljivostih. Pri editorju pogrešamo ukaz RENUMBER za avtomatsko preštevilčanje vrstic, pri monitorju pa enokoračno izvajanje programa, saj je odkrivanje napak pri tem neprimerno lažje.

Če ste se torej naveličali basica, pascala in drugih »višjih« programskih jezikov, pojdite v banko, nakažite 60 funtov na spodnji naslov in priložite bančno potrdilo kratkemu pisemcu, naslovljenemu na še bolj spodnji naslov (navodila so namenjena tistim, ki imajo devizne račune, pa ne vedo, kaj naj z njimi počnejo). Čez slabih štirinajst dni boste imeli v rokah orožje, s katerim boste prisilili trmoglavo napravo, da vam bo izdala vse svoje skrivnosti.

Naslov banke:
National Westminster Bank,
381 UX Bridge Road,
Hatch End, Middlesex
Code: 60-10-16
A/c: Supersoft
No: 38046385

Naslov podjetja:
Supersoft
Winchester House, Canning Road
Wealdstone, Harrow
Middlesex, HA3 7SJ
United Kingdom

Vizawrite 64

DUŠAN ARKO

Vizawrite 64, doslej najboljši program za urejanje besedil s commodorom 64, odlikujejo razkošna grafična oprema, smotrni ukazi in velika hitrost. Hišni računalnik se z njim in s programom Vizastar 64 (računanja v stolpcih in vrsticah, zbirka podatkov in poslovna grafika) dvigne v razred osebnih računalnikov.

Program Vizawrite 64 ni zanimiv samo za zasebnika ali tistega, ki poklicno dosti piše, temveč tudi za majhne podjetje, ki hočejo dobiti za malo denarja dober računalniški sistem. Za osebne računalnike je treba odštati nekajkrat več (v ZR Nemčiji od 2000 DM navzgor), pri tem pa se končni produkt, narejen z Vizawriteom 64, v ničemer ne loči od tistega, ki je nastal s PC. Tudi hitrost in udobnost sta primerljivi.

Urejenje besedile

Program se dobi na disku ali v modulu. Po vpisu z diska v pomnilnik oziroma po vključitvi računalnika se na zaslonu prikaže glavni menu, ki ponuja različne možnosti. S tipkami F1–F7 lahko izberete pisanje novega besedila, izpis starega, izpis seznama zapisov na disku ali ukaze za disk.

Če se odločite za pisanje novega besedila, dela računalnik v modusu urejanja besedil. Na zaslonu kazeta prva in tretja vrstica naslov besedila, stolpec in vrstico, v kateri je kurzor, številko strani in število vseh strani besedila. V drugi vrstici se vedno prikažejo ukazi, npr. »zapis besedila«. V zadnji vrstici so opozorila na morebitne napake.

Pomnilnik s 33 000 znaki je za to kategorijo dovolj velik (približno deset tipkanih strani formata A 4). Prednost tega programa pred drugimi je, da lahko ukaze uporabljate med pisanjem besedila, ne da bi vam bilo treba spremeniti modus.

Preden začnete pisati, lahko nastavite tabulatorje. Robna dva (levi in desni) sta že naravnana na 5 in 74 stolpec. Tabulatorje lahko premikate do 240 stolpca. Pišete kar naprej, ne da bi pazili na desni rob kot pri običajnem pisalnem stroju. Na zaslonu vidite naenkrat samo 38 znakov, ko pa pridete z besedilom oziroma s kurzorjem na konec zaslona, se »okno« samo premakne na desno (horizontal scrolling). V nasprotju z drugimi programi za urejanje besedil Vizawrite 64 ne odreže besede, ki ne gre več v vrstico, temveč jo da v naslednjo vrstico (word wrap). Navadno se piše osnutek brez preloma, pri tem pa se besedilo na vsaki dve strani zapiše na disk, da bi pri morebitni izključitvi električnega toka zgubili vsega.

Ste kaj pozabili? S kurzorjem greste na mesto, kjer hočete vstaviti manjkajoče besedilo. S tipko F7 razprete na tem mestu vrstico in pišete vmes, kolikor vas je volja. Z isto tipko priključite drugo besedilo.

Čudite se hitrosti, s katero računalnik izveda vse ukaze. Commodore 64 pri tem delu kar oživi in nič več ne spominja na strojček s počasnim basicom. Podobno je pri brisanju. S kurzorjem greste na začetek besedila, ki ga hočete zbrisati, pritisnete na tipko F8 in premikate kurzor po besedilu. To postaja svetlejše, s tipko RETURN pa ga čisto zbrišete. Pri manjših popravkih seveda lahko uporabljate tipko INST/DEL. Vsak ukaz lahko prekinete s tipko STOP.

Vsi ukazi za izpis besedila s tiskalnikom se vidijo tudi na zaslonu. Nekaj jih je že vdelanih v Vizawrite: poudarjana pisava, podčrtanja, indeksiranja in potenciranje. Ti ukazi imajo simbole, ki jih prej vstavimo v besedilo s tipko CTRL. Poleg tega lahko v oblikovani vrstici (format) s tipko CTRL (investirani znaki) in s številkami od 0 do 9 definirate ukaze za tiskalnik, ki se v besedilu ponavljajo. Pri tiskalnikih, ki imajo sekvence ESC, zadostuje, da po številki ukaza navadete črko sekvence. Pri Epsonovem tiskalniku FX 80 je ukaz v oblikovalni vrstici za proporcionalno pisavo takle: 0=p 1=1 (CTRL+0=p CTRL+1=1).

Z Vizawriteom 64 lahko pišete tudi serijska pisma. Naslovi oziroma besedilo z »delovne strani« (work page) bodo vpisani povsod tam, kjer je znak MERGEK. Na to stran lahko sami vstavite besedilo ali pa ga nanjo izpišete iz datoteke. Z navedbo v rubriki Global/Fill v menuju za tiskalnik bodo ti podatki vstavljeni na pravo mesto v besedilu.

V besedilo lahko nanizate poljubno število besedil z diska. Edina omejitev je zmogljivost diskete.

Urejati je mogoče tudi zapise, napisane z drugimi urejevalniki besedil (npr. Wordpro, SM Text, Easyscript), sekvenčne datoteke, seznam zapisov na disku in celo datoteke iz programa Busicalc.

Tiskenje

Korekturi osnutka na zaslonu sledi izpis s tiskalnikom. V ta namen ime Wizawrite 64 poseben menu. Ker je v program že vdelan Centronicsov programski vmesnik, ni nobenih težav s priključitvijo tiskalnikov s tem vmesnikom (npr. FX 80). Poseben izdatek je samo kabel, ki si ga pa lahko naredite tudi sami; material stane približno 50 DM. Pri tiskalnikih CBM, npr. MPS 801, MPS 802, ta kabel seveda ni potreben. Celo barvni risalnik in tiskalnik CBM 1520 lahko uporabljate, le njegov naslov je treba spremeniti v 5.

Tiskanje lahko vedno ustavite in nadaljujete ali pa začnete znova. Možno je tiskanje posameznih listov. Poleg tega lahko izberete dolžino papirja (pri »neskončnem« papirju formata A 4 in dolžina 30,5 cm, 72 vrstic), levi rob, tiskanje v bloku (zravnava desnega roba), razmik med vrsticami in različne oblike črk. Na vrhu strani lahko vstavite glavo (header), ki je potem za vse strani enaka.

Ko je osnutek na papirju popravljen, naredite prelom tako, da na vsakih 60–64 vrstic postavite znak za konec strani (CTRL+p). Na vrsti je drugo in v večini primerov tudi zadnje tiskanje besedila.

Ocene

\+ Neposredna priključitev tudi drugih tiskalnikov, ki imajo Centronicsov vhod (npr. Epson) prek kabla na uporabnikova vrata. Ne potrebuje nobenega hardwarskega vmesnika.

\+ Piše tudi serijska pisma (okrožnice).

\+ Na besedilo se lahko naniza več besedil ali datotek z diska.

\+ Izpisuje in ureja besedila drugih urejevalnikov besedil.

\+ Lahek in hiter prelom strani.

\+ Tiskanje v bloku (zravnanje desnega roba) s samo enim ukazom.

− Precej drag je (pribl. 300 DM).

− Samodejno oštevilčenja strani se ne da poljubno nastaviti.

− Jugoslovanski znaki zaenkrat še manjkajo.

Ukazi za urejanje besedila:	
CBM+c	kopira besedilo na drugo mesto
CBM+m	prenese besedilo na drugo mesto
CBM+d	ukazi za disk
CBM+f	poišče del besedila
CBM+F	poišče del besedila, ne glede na male ali velike črke
CBM+g	skoči na želeno stran
CBM+M	izpiše besedilo na disku v besedilo (marge)
CBM+n	spremeni naslov besedila
CBM+p	izpiše besedilo s tiskalnikom
CBM+q	nazaj v glavni menu
CBM+r	nadomesti besedo
CBM+R	nadomesti besedo v vsem besedilu
CBM+s	zapiše besedilo na disk
CBM+t	nastavi barvo besedila, ozadja in okvira
CBM+w	prikaže besedilo v širini zaslona
CBM+x	zapiše besedilo na disk ali kaseto
CBM+$	izpiše na zaslon seznam zapisov na disku
CBM+DEL	zbriše besedilo
CBM+INST	vstavi besedilo
CBM+SPACE	pokaže število prostih celic v pomnilniku
Ukazi s tipkami F1–F8 HOME in CLR (pri urejanju besedila).	
F1	stran naprej
F2	stran nazaj
F3	zaslon naprej
F4	zaslon nazaj
F5	kurzor na naslednji tabulator ali konec vrstice
F6	kurzor na začetek vrstice
F7	razmakne besedilo za vstavek in ga spet priključi
F8	zbriše besedilo
HOME	pri prvem pritisku skoči kurzor na zgornji del zaslona, pri drugem pritisku pa na začetek besedila
CLR	kurzor skoči spet na konec besedila
Ukazi za oblikovanje:	
CTRL+i	premakne besedilo (indent paragraph)
CTRL+t	tabulator
CTRL+	številčni tabulator
CTRL+c	centrira besedilo v vrstici
CTRL+p	konec strani
Ukazi za tiskalnik:	
CTRL+u	podčrta
CTRL+e	indeksira (subscript)
CTRL+↑	potencira (superscript)
CTRL+a	poudarjeno pisanje
CTRL+#	oštevilčenje strani
CTRL+0	CTRL +9 odda kodo ASCII

Prijetno presenečenje

Ivan Bratko, Vladislav Rajkovič: Računalništvo s programskim jezikom pascal

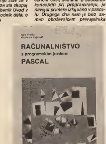
PRIMOŽ JAKOPIN

Tako rekoč brez vsakega hrupa, brez predhodne nekajtedenske »priprave« v javnih občilih, se je na policah naših knjigarn znašla še ena knjiga s področja računalništva, učbenik Računalništvo s programskim jezikom pascal. Namenjena je srednješolcem, izdala jo je Državna založba Slovenije, napisala pa sta jo Ivan Bratko in Vladislav Rajkovič. Avtorja obema segajo korenine na fakulteto za elektrotehniko, sta uveljavljena in priznana strokovnjaka: prvi za umetno inteligenco, drugi za računalniško podprto odločanje (»computer aided decision making«). Po njuni zaslugi segajo začetki na področju uvajanja računalništva v srednje šole že v leto 1970, leta 1974 pa sta skupaj izdala standardni učbenik Uvod v računalništvo, predhodnik dela, o katerem teče beseda.

Drugi del je posvečen paradnemu konju evropske računalniške znanosti sedemdesetih let, višjemu programskemu jeziku, ki se imenuje po znanem francoskem matematiku in filozofu. Gre za temeljit prikaz, ki se ni ognil zahtevnejšim konstruktom – zadnji poglavji opisujete rekurzivne podprograme in strukture s kazalci. Vprašanje, zakaj ravno pascal in ne kaj drugega, v delu ni bilo deležno posebne pozornosti, je pa nanj res lahko odgovoriti.

FORTRAN, ki ja bil naveden kot glavni programski jezik v »Uvodu« iz leta 1974, prav gotovo ne pride več v poštev. Res je bil osnovni jezik v emeriških in naših univerzitetnih okoljih šestdesetih in sedemdesetih let (starejše generacije programerjev se ker ne morejo posloviti od njega), res je pa tudi, da ga je čas povozil.

Ko je imel profesor Bauer z univerze v Zürichu, svetovne avtoritete, v okviru simpozija ,Informatica 74' na Bledu (kjer so že tisti sončni časi) seminar o sodobnih konceptih pri programiranju, je navajal primere izključno v pascalu. Drugega dne nam je bilo samim oboževalcem prevajalnika



Knjiga je sestavljena iz dveh delov, iz 129 strani dolgega uvoda in iz obravnave programskega jezika pascal namenjenega drugega dela, ki sega do strani 295. V uvodu, ki obsega sedem poglavij, se seznanimo z osnovnimi pojmi ter zgradbo in delovanjem računalnika, nato pa najdemo še nekaj o programskih jezikih, računalniških sistemih, razvoju algoritmov in uporabi računalnikov. Snov je skrbno obdelana, razpored velikokrat preizkušen v šoli, jezik pa tudi ni slab. Govorjenja v prazno v njem ne bomo zasledili.

formul, po malem dovolj in se je nekdo le ojunačil ter vprašal: »Kaj pa FORTRAN?« Odgovor je bil kratek in uničujoč.

C, ki je sinteza lepo strukturiranega višjega programskega jezika in strojnega jezika in ki ne tujem v zadnjem času neustavljivo napreduje na mikroračunalniški fronti, je pri nas za večino še uganka: ustrezni prevajalniki na široko razširjenih računalnikih so v obtoku šele dober mesec dni. Strojni jezik sam moramo seveda tudi zavrniti – res je najučinkovitejši in omogoča cel kup mikrosekundnih vragolij, je pa tudi res, da je jezik sužnjev, stroju veliko bližji kot človeku in zato za osnovno šolsko rabo docela neprimeren.

Basic bi bil tako videti nekako edini resen protikandidat – je pač materin jezik večine ljudskih računalnikov in vsakomur takoj pri roki. Je pa v resnici najmanj eleganten in čepi na socialni lestvici programskih jezikov popolnoma upravičeno še nekaj klinov pod fortranom. Že narečje v Sinclairjevi Mavrici je precej skromno in kakšnega strukturiranega sprenevedenja ne prenese, pravč pri Commodorjevem uspešnežu pa je položaj še za pol konja slabši. Bistveno boljše možnosti za vključitev basica v učbenik bi ponujal, če bi prodrl v naše šole, Sinclair QL: njegov SUPERBASIC podpira veliko sodobnih krmilnih struktur, pozna procedure in rekurziven je. Žal tudi ta možnost najbrž odpade: QL-i so pri nas kot pamet v Butalah.

Če se na koncu spet vrnemo k učbeniku – v okoliščinah, kakršne so pri nas, pomeni ,Računalništvo s programskim jezikom PASCAL' veliko in pomembno pridobitev za naš knjižni trg. Založbi gre pa tudi vse priznanje: knjiga je kljub tristo stranem, kljub kvalitetnemu in jasnemu tisku zelo dostopna – stane samo 671 dinarjev.

ABC v drugo

CIRIL KRAŠEVEC

Knjigo ABC računalništva, ki jo je izdala Zveza organizacij za tehnično kulturo Slovenije, smo podrobneje predstavili že v drugi številki našega časopisa. Pred približno mesecem dni pa je knjiga istega avtorja izšla še v luksuznejši izvedbi.

ABC računalništva partner HR-84, kot je pravi naslov druge izdaje, je plod sodelovanja ZOTKS in Iskre Delta. V zbirki Delta je knjiga tudi brez tehničnih pomanjkljivosti. Tisk je normalen in čist, platnice so trde in plastificirane. Ilustracije so iste, vendar zaradi nove tiskarske tehnike lepše in bolj pregledne. Tudi pri jeziku je precej bolje, saj je pri zelo na hitro narejeni prvi izdaji kar mrgolelo napak, ki so tukaj vestno odpravljene. Da pa le ne bo ostalo samo pri ponovni izdaji in čistem Iskrinem dobrotništvu, je prišlo v knjigo nekaj dodatkov.

Popolnoma novo je osmo poglavje, namenjeno predstavitvi mikroračunalniškega sistema partner, ki bo morda kdaj našel mesto tudi na vaši mizi. Informacija je zanimiva, vendar v takšni obliki nikakor ne sodi v knjigo. Krepko nam je jasno, da bo »partner vaš resnični sodelavec pri delu« samo, če ga boste kupili. Cena pa nikakor ne sodi v rang ABC.

Druga sprememba, čeprav še vedno reklamna, je zelo dobrodošla za mlade programerje, ki bodo hočeš nočeš v šolah pisali svoje prve programe s HR-84. Prvi koraki v programskem jeziku basic so v tej izdaji aplicirani prav na ta Iskrin računalnik. Pregledani so vsi ukazi basica in ilustrirani s primeri. Navedene pa so tudi procedure pri vklopu in operacije v monitorskem programu, da pridemo do basica.

Knjiga je vsekakor dobrodošla na knjižnih policah, saj jo bo kupil marsikateri začetnik, ki mu je prav izdaja ušla pred nosom ali pa takrat še ni mislil na znanost, ampak le na video igre. ABC računalništva v novi preobleki ne stane dosti več kot v prvi izdaji: prodajajo jo po 500 din.

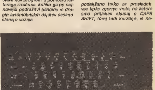
Naslovnico slovenski bralci že poznajo: to je bila naslovnica septembrske številke, ki smo jo brž razprodali. Zdaj smo jo uporabili še za prvo številko srbskohrvatske izdaje Mojega mikra, ki bo izšla sredi decembra, od februarja pa bosta izdaji vzporedni in skoraj enaki. Nova spodbuda za avtorje prispevkov, sestavljalcev programov in naročnike malih oglasov: vsa ta gradiva bodo še odmevnejša, saj bodo prišla v roke vseh jugoslovenskih hackerjev.

●

Jugoslovanski revialni tisk je zadnje tedne poln obširnih člankov o razcvetu YU računalništva. Znani zagrebški tednik Danes je najprej dokumentirano opozoril na strahovito zaostajanje v elektronski industriji, potem pa je pod naslovom informacijski pohod na kioske seciral vse štiri glavne publikacije, ki se na jugoslovanskem trgu ukvarjajo z računalniško problematiko. Moj mikro si je priboril samostojno barvno sliko prve naslovnice in o njem je zapisano, da je poln »mikroračunalniške prakse in programskih listingov, in sicer najpotrebnejših, ki so namenjeni sinclairu, medtem ko o večjih mikroračunalniških zverinicah, kakršni so macintosh, HP 150 in iskrin HR 84, govori z odmaknjenostjo, kakršno zahtevata zdrava pamet in njihova cena«. In še: »Moj mikro zaradi svoje lažernosti (igre, grafika, rubrika o hackerjih, zapisi po zgledu onega o osebnih robotih) očitno ne želi zaiti v nevarnost (ki bi utegnila ogroziti BIT), da bi potreboval dotacijo.« Danes sicer ocenjuje BIT kot »računalniškega aktivista«, za Politikin Svet kompjutera pravi, da je »mikroračunalniški tabloid«, ki pa bo moral v svoji »prijatni noriji poiskati kak sistem«. Računari u vašoj kući pa se mu zdijo kljub praktičnemu okviru preveč zaznamovani z »ono že tako zelo demodirano znanstvenofantastično vznesenostjo«, ki je »logična posledica hitrega preskoka iz opank k tastaturi«.

●

Jože Smole, predsednik mestnega komiteja ZK Ljubljane in član CK ZKS, je zgleden – lahko bi rekli celo enkraten – primer politika starejše generacije, ki mu računalništvo ni tuje. Ne bere samo Mojega mikra (z lupo, ker nima več hackerskega vida in se pritožuje, da so Listi preveč pomanjšani), temveč tudi sam sestavlja programe. Na Gorjupove dnevu veliko srečanje slovenskih novinarjev v Radencih je prišel s Sharpovim mikroračunalnikom v prsnem žepu. V njegovem spominu ima shranjena kode, s katerimi lahko prikliče na primer podatke o tem, kje so se mudili, se mudijo in se bodo mudili njegovi politični kolegi. Med govorom Franca Šetinca pa je Jože Smole pridno sestavil nov program, s pomočjo katerega izračuna, koliko ga po najnovejši podražitvi bencina in drugih avtomobilskih dajatev cesarju stane vožnja.



Tudi Sinclair je v pričakovanju povečanega novoletnega povpraševanja predstavil nekaj »novega«: ZX spectrum 48 K v preobleki QL. Kakšen bi bil profil uporabnikov ZX spectruma, če bi že od vsega začetka imel tako tipkovnico, lahko le ugibamo. Tako, kot je sedaj, omogoča resno delo tudi tistim, ki ga domov niso prinesli skritega za pasom. Potencialni kupci naj bi bili iz vrst tistih, ki se ravno zaradi neugledne tipkovnice niso odločili za spectrum.

O novem modelu so nas površno obvestili že med našim obiskom v Angliji. Priznati pa moramo, da smo pričakovali več. V bistvu pa je nova le tipkovnica. Kot vidite na sliki, močno spominja na tisto, ki je vdelana v QL, torej le ni čisto taka, kot so druge profesionalne tipkovnice. K sreči ima nekaj tipk več kot stari spectrum: podaljšano tipko za presledek, vse tipke zgornje vrste, na katere smo pritiskali skupaj s CAPS SHIFT, torej tudi kurzorje, in nekaj najvažnejših ločil. Razporeditev ni najboljša, morda bi kazalo nekaj tipk z leve postaviti na desno, tako da bi bili pike in vejica v isti vrstici.

Novost je še tipka za reset. Interface 1 se privije s spodnje strani, tako kot pri starem spectrumu.

In cena? Približno toliko dražji je, kot če bi kupili novo tipkovnico: 160 funtov stane. V ZR Nemčiji naj bi stal približno toliko kot C-64, okrog 700 DM. Sinclair s spec-

Revija Sinclair User je v novembrski številki posvetila eno ekorajda glavnih tem, strip o Sinclair Simonu, Jugoslaviji. Spremlja ga tudi zapis o računalništvu pri nas, da smo pretihotapili 20 000 spectrumov, da je s tem nastala igra Kontrabant in da na gospodarski zbornici pravijo »Prepovedi uvoza ni«.
Risar stripa pa očitno ni obveščen o cenah računalnikov pri nas.

Moža s polno torbo spectrumov, ki jih je uspešno spravil mimo carinika, bi na ljubljanskem bolšjem trgu prav hvaležno sprejeli, saj smo prepričani, da si ne bi vzel 200 odstotkov profita. Avtor nam je v telefonskem pogovoru zaupal, da so v torbi spectrumi 48 K, ki jih v Jugoslaviji še ne »delajo«.
Angleži se smejejo, mi pa kislo mežikamo.

trumom torej še vedno dosega visoko ceno, ker kaže, da zvezda ne temni

●

Škotska univerza iz Edinburgha je razvila programski komplet Expert-Ease, za katerega trdijo da omogoča uporabo umetne inteligence v mikroračunalnikih No vost je sestavljena iz programov, shranjenih na gibkih diskih za razne vrste računalnikov, in priročnika Po vstavitvi gibkih diskov sistem avtomatsko vodi pogovor s svojim učiteljem, strokovnjakom s kakšnega področja Dokaj naglo in enostavno se »nauči« vse parametre, potrebne za proces razmišljenje in odločenja o problemih Uporabljajo se različni primeri možnih odločitev tako da lahko sistem kar najbolj netančno določi algoritma, s katerimi bo pozneje obdelal enak problem in pomagal pri odločitvi tudi uporabniku nestrokovnjaku

Pri vnašanju informacij o postopkih odločitve (»učenju«) sistem sam podrobno izrazi pravile in opozarja na dejavnika ki se ne ujamejo z že vnesenim znanjem, oziroma na primere ki so protislovni, pa tudi na taka, za katera so podatki že shranjeni S takšnimi povratnimi informacijami lahko uporabnik hitro in netančno sestavi kompleksna modele revnenja v določenih situacijeh

Expert-Ease je uporaben v skoraj vseh znanstvenih disciplinah, povsod tam kjer je mogoče strokovno znanje vdelati v sistem in ga dati pozneje ne voljo nestrokovnjakom Hkrati se da z njim preverjati in dopolnjevati znenje strokovnjakov Najbolj učinkovit je seveda pri diagnostiki napak v kompleksnih sistemih in seveda medicini, neki proizvajalec helikopterjev pa je s tehniko Expert-Ease podpri široko mrežo svojih servisov

Žiga Turk, Matevž Kmel, Jani Kovačič in Sašo Jankovič končujejo projekt, ki so ga pripravljali zadnjih pet mesecev Risano pustolovščina Kontrabant II bo izdala Založba kaset in plošč RTV Ljubljana igra je logično nadaljevanje Kontrabanta I, v katerem naj bi si prilihotapili računalnik. Če ga, niste nič hudega, v Kontrabantu II ga dobite brez muk. Kaj z njim početi, to je zdaj vprašanje Prve korake bo začetniku olajšal okrog 30 strani dolg priročnik. Program je več kot samo igra, saj se boste na poti skozi čas (geslo je »Z računalnikom pod pazduho v 21. stoletje«) morali naučiti marsičesa o zgodovini (in zemljepisu) Jugoslavije.

Bralec Bogdan Ševnik iz Ljubljane nam je poslal naslednjo travestijo

OD TERMINOLOŠKE PRAVDE

Poet naš nov besedni venec vije
Misli starih je spomin in njege hvale
Razvoj je šel naprej, metode stera je ostale
Važna je beseda, pa čeprav ne sije

Al prav piše se izložek al izhod
To zdaj mori Slovenca brata!
Vsemu drugemu smo brž zaprli vrata
Postal je vložek vhod da le ni besednih zmot

O varujmo se razuma abstinence!
Hranimo moči za tam kjer nam jih bolj je treba!
In ne pravdajmo se spet okoli oslove sence!

Ne zameri nam besedo pikro
če resnico više ceniš kakor hvalo!
Le resnici služit hoče le
MOJ MIKRO!

F (ne več) Prašerni

●

Društvo znanstvenih in tehniških prevajalcev Slovenije je sredi novembra priredilo tridnevni seminar Računalniška terminologija v praksi. Največje veselje je pripravil interni slovar tehniških izrazov (prevodi iz angleščine v slovenščino, srbohrvaščino in ruščino ter iz slovenščine v druge tri jezika), ki so ga 30 oktobra računalniško izpisali v programskorazvojni službi Iskre Telematika, predstavila pa ga je prof Marta Kadunc. Društvo znanstvenih in tehniških prevajalcev je na okrogli mizi o jeziku in računalništvu na Slovenskem napovedalo, da bo do pomladi sestavilo svoj slovar V Mojem mikru, kjer smo na prvi strelni črti angleško mislečih hackerjev, ga včasih krvavo potrebujemo.

V začetku bi rad pohvalil in pozdravil vašo revijo. O tem so dovolj napisali že bralci pred mano, zato rečem le, da se z večino strinjam tudi jaz.

Pišem zaradi nasveta. Sem namreč lastnik spectruma (od nedavnega), na področju računalništva pa sem popoln začetnik. Obstaja pa volja in prvi skromni koraki so tudi že za mano. V drugi številki revije ste objavili program LSD Žige Turka. Ob njem sem izgubil že mnogo ur (nočnih) in dosegel celo rob družinskega spora, pa ne gre, ne dela. Program bi mi prišel prav pri urejanju programov. Zato bi prosil za podrobnejšo razlago listinga 2 in sestavljanja obeh delov programa (dopolnilo iz št. 4 ne funkcionira), poleg tega pa:

1. Katera je spremenljivka CL?
2. Zakaj ne dobim na zaslon listinga 2, ko vpišem tabelo, oz. če ga dobim, ne morem ničesar zbrisati, pa tudi vrstice 1000 ne morem spremeniti v 1?
3. Kako lahko popravim napako, če se zmotim pri vnosu v tabelo?
4. Zakaj se mi pojavljajo razna sporočila: Invalid colour, Ramtop itd. (ne pa Out of memory)

Za konec me zanima, ali je še kje mogoče kupiti kaseto RŠ s programi za spectrum (LSD ne bi uporabljal za kopiranje programov s te kasete). V trgovinah po Ljubljani sem jo iskal, pa je že razprodana.

Želim vam še veliko uspeha pri delu.

Miloš Paternel,
Beblerjev trg 9,
61110 Ljubljana

Vsi novinci v računalništvu se praviloma najprej srečajo s programi za presnemavanje. Zakaj, nam je še vedno uganka.

Sistem kirurških posegov na koncu je združiti stavek REM iz programa 2 z glavnim programom, potem pa začetek basica prestaviti čez vrstico REM. Če boste natančno sledili navodilom, se vam mora posrečiti.

1. CL, kot je razvidno iz listinga, je napisana v stavku DATA v vrstici 42. READ v vrstici 43 jo prebere in naredi CLEAR. Če jo povečate, napake iz vprašanja 4 ne bi smelo biti.

2. Edini odgovor na vaše vprašanje je: napačno zapisana vrstica 1100 ali pa prekratek REM.

3. Tabelo popravite s POKE, naslov, pravilna vrednost.

4. Kasete s programi ta hip ni mogoče kupiti, vendar bo kmalu ponatisnjena. Z malo sreče boste našli srbokrvaško verzijo.

Kadar se pojavljajo napake, ste nekaj napisali premalo natančno.

Verjetno ste LIST ali popokali REM.

Pišam vam ob našem računalniškem prebujenju. Računalniška revija Moj mikro je pokazala, da se je nekaj vendarle začelo premikati. Res je zasnovan na visoki ravni in bo prišel prav vsakemu začetniku in hackerju. Je pa tudi programsko dobro zasnovan. Začetnik najde veliko programov iz katerih zve pomen programskih jezikov.

Toliko o tem. Imam pa še problem v zvezi z Radiom Študent. Vem, da so v softwerski redakciji sami »brihtni tipi« in znajo dobro zaščititi programe, ki jih potem predvajajo. Vendar program, ki ga posnameš, drži oz. se da včitati le kakšen teden. Če hočejo »mojstrom« bistriti glavo drugim, lahko dajo poseben zaščiten program npr. ob 12. uri. Tako je s presnemavanjem. Šepa pa tudi kvaliteta posnetka, tako da so problemi s presnemavanjem oz. nasploh z včitavanjem. Za informacijo naj povem, da stanujem v Mostah in nimam slabega kasetofona.

Drugič pa se lahko delo RŠ le pohvali. Slišal sem, da bo kmalu izšla nova kaseta za ZX spectrum, Moj mikro pa bo preveden v srbohrvaščino. To je v redu.

V kratkem, če se bom potrudil, bom poslal najnovejšo igro posebne vsebine, za katero upam, da je v redu. Jasno, moj izdelek. Poleg tega pa imam kup neuresničenih idej.

Veliko sreče pri urejevanju revije!

Zlatko,
Ljubljana

Problemi pri snemanju programov RŠ največkrat nastanejo zaradi motenj, ki se radijskemu signalu pridružijo na poti med oddajnikom in sprejemnikom. Motnje, ki jih med običajnim poslušanjem slišimo kot pokanje in brnenje, nastanejo zaradi nepravilne priključitve električnih porabnikov ali zaradi bližine neblokiranih avtomobilov. Običajno nas to ne moti preveč, za tonski zapis programa pa je usodno. Takšnih motenj v principu ne moremo odpraviti, zato vsak program na RŠ oddajajo večkrat. Če ne gre prvič, torej poskusi drugič!

Pozdravljeni Mikrovci!

Oglašam se vam iz ZDA, zvezne države Washington. Za eno leto sem zamenjal bivališče (sem dijak četrtega letnika), to pa ne velja za računalniško revijo – še vedno sem zvest Mojemu mikru, ki je brez pomislekov najboljša jugoslovanska revija! Posebej je treba pohvaliti zasnovo. Vsakdo lahko najde kaj zase. Skratka, revija je

Ubogi Jet Set Willy!

Načrt sob, ki ga je za prejšnjo številko narisal BRANKO NOVOSEL iz Pitomače, je razkuril ljubitelje Jet Set Willyja. Izkazalo se je, da mora možiček v igri prehoditi še več neverjetnih sob, kot pa nam jih je predpisal naš mladi dopisnik. Do zaključka redakcije (21. novembra) so se nam s popravki oglasili štirje bralci. Če bo kdo odkril še kakšno sobo, naj nam piše!

V načrtu manjka soba, ki jo je precej težavno najti (tudi sam sem se dokaj namučil). To je soba The Forgotten Abbey, edini izhod iz Wine Cellarja. Vanjo pridemo tako, da stopamo z zravnanimi nogami k stopnicam v spodnjem desnem kotu in hkrati pritisnemo tipki za skok in premikanje na desno. Če se vam bo posrečilo, boste na pravi poti. Iz te sobe (zelo težavno je priti skoznjo) boste stopili v sobo Security Guard.

Marko Rukonić,
Nazorove 21,
Zagreb

Opazil sem, da ima moj načrt več sob, se pravi, da vam je Branko Novosel poslal pomanjkljiv načrt JSW. Jaz sem odkril še tri sobe, skupaj 60 sob.

Prva je desno od Wine Cellarja in se imenuje The Forgotten Abbey. Vanjo se pride iz Wine Cellarja.

Druga je pod sobo The Drive in se imenuje Under the Drive. V to sobo se pride iz Wine Cellarja, vendar vmes preskočiš šest sob. Najprej prideš v The Security Guard, a ne zgoraj na stopnicah, temveč čisto spodaj. Nato preskočiš luknjo, greš naprej po robu in že si v tej novi sobi.

Tretja je pod sobo The Foot of the Megatree in desno od sobe Under the Drive, imenuje se pa Tree Root. Pot vanjo je ista kot v Under the Drive, le da prehodite to sobo in greste v novo.

Našel pa sem še nekaj. Ko pridete (od spodaj) v The Security Guard in se nato vržete v luknjo, se znajdete v sobi, v kateri je samo velik napis DIE MORTAL. To sobo samo preletita, nato pa padete v The Off Licence in se tam dokončno ubijete, tudi če imete še vsa življenja.

Na koncu naj pohvalim vašo revijo, ki mi je zelo všeč, še posebej članki o novih igrah, Vaš mikro, programi itd. Na splošno všeč mi je od prve do zadnje strani. Le pod napisom Moj mikro na naslovni strani naj čim dlje ostane številka 200.

Še veliko uspeha pri urejanju MM vam želi vestni bralec

Dejan Jovanović,
Tabor 6,
Ljubljana

S primerjanjem z mojim načrtom sem ugotovil, da načrtu v reviji manjkajo še štiri sobe, ki sem jih sam odkril. Zato vam pošiljam načrt teh štirih sob in upam, da ga boste objavili v korist vnetih igralcev te igrice.

Sobe so The Forgotten Abbey, Entrance to Hades (tu zgubiš vsa življenja), Under the Drive, Tree Root. Ko v sobi Wine Cellar pridete čisto v desni spodnji kot, skočite v zrak, tako da boste pristali na sredini stopnic pod takojšnjim trakom. Nato odidite ravno, dalje skozi svetlo modra atributa v zidu sobe in prišli boste v The Forgotten Abbey. Najprej prepuščam presenečenje vam.

Petar Kovačić,
Dobravska 21,
Maribor

primerne za začetnika na računalniškem področju kakor tudi za mojstra programiranja. Pohvalil bi rad rubriko Naloge za hackerje, ki jih zelo rad rešujem. Vsa pohvala tudi prvi slovenski računalniški kaseti, ki je po mojem mnenju dobra razen nekaterih malenkosti (angleško-slovenski slovar, ki še ni dokončan). Dobili smo ravno na katero smo dolgo čakali!

Kar zadeva »mavrice«, ki jih je uvozila Iskra, pa sem ostal brez besed, ko sem slišal za ceno! Ne vem, zakaj takšne cene – za ta denar lahko v Nemčiji kupite dve »mavrici« in še kakšen dodatek! Se potem sploh lahko čudimo, ko slišimo kako se »ilegalno« uvažajo »mavrice«? Mislim, da se na tehničnem področju ne bi smeli iti takšnih nezdravih igric, kajti to se nam bo še krepko maščevalo (če se nam že ne). Vse, kar lahko naredim, je to, da svetujem tem možakarjem, kio so si izmislili takšne vrtoglave cene, naj se pozanimajo, koliko stanejo te reči v tujini!

Še novica za ljubitelje breaka. Pred kratkim so v New Yorku predstavili program za commodora 64 Break Dance »Plesalca« krmilite z igralno palico in zmore večino najzahtevnejših likov, ki jih lahko vidimo po naših diskotekah. Program se odlikuje z dovršeno grafiko, ki je v veliki meri odgovorna za njegov uspeh.

V novembrskem Mikru je objavljenih 57 sob, od tega je Emergency Generator narisan dvakrat. Manjkajo 4 sobe, skupaj jih je 60. Res je, da se iz sobe Rescue Esmeralda pride navzdol v Emergency Generator, vendar zaradi tega ni treba narisati sobe dvakrat. Soba, kjer se začne igrati, ni Bedroom, ampak Bathroom.

Nekatere sobe so nekoliko premaknjene glede na skico v Mikru. Tako nastane prazen prostor med sobama The Attic in Under the Roof. Prav tako Priests Hole ni nad sobo Half Way up to East wall.

Pomembnejše so štiri sobe, ki so na objavljeni skici izpuščene. Entrance to Hades obstaja, vendar je nesmiselno iti vanjo, ker se igra konča. Tudi nobenih predmetov ni tam. Ni pa tako z drugimi tremi lokacijami – The Forgotten Abbey, Under the Drive in Tree Root. Tam je dosti predmetov.

Skok, ki prav tako ni narisan, je iz sobe Rescue Esmeralda v Ballroom East. Možno je, da je še kak podoben skok.

Upam, da boste pomanjkljivosti popravili.

Iztok Stražar,
Kajuhova 44,
Ljubljana

Med najbolje prodajanimi osebnimi računalniki je v ZDA na prvem mestu commodore 64. Stane 195 dolarjev, tiskalnik 240 in diskovni pogon pa dobite za 220 dolarjev.

Marjan Truci,
Walla Walla,
Washington, ZDA

Časopis moj mikro mi je zelo všeč. Če je mogoče, poskusite v eni od prihodnjih številk posvetiti nekaj strani računalniški grafiki. Imam spectrum in nikakor ne moram vpisati znaka na matriko 16×16. Zanima me, kako naj to dosežem.

Vladimir Kanovnik,
B. Maslarića 4/V,
Zagreb

Program za povečavo črk najdete v tokratni programski prilogi, pomagate pa si lahko tudi z naslednjim prispevkom, ki nam ga je poslal bralec **Borut Lenardič** iz Ljubljane.

Odlična Psionova rutina za povečavo črk na kaseti Horizons je prilagojena spectrumu 16 k. Začne se na naslovu 32256 in obsega 300 znakov. Pri verziji 48 k je v kombinaciji z daljšimi programi v basicu ne moremo uporabiti, jo pa lahko preprosto predelamo in prilagodimo shranjevanju na najvišjih naslovih pomnilnika 48 K.

Naložimo katerikoli program s strani B kasete Horizons, ga prekinemo z Break in vnesemo naslednjih 5 ukazov:

```
POKE 32343 251
POKE 32363 251
POKE 32384 251
POKE 32411 251
POKE 32506 251
```

Zdaj shranimo rutino na trak:

```
SAVE »ime« CODE 32256,300
```

Verificiramo, izklopimo in vklopimo računalnik ter ponovno naložimo rutino:

```
LOAD »ime« CODE 64256,300
```

Ponovno jo shranimo na trak:

```
SAVE »ime« CODE 64256 300
```

Rutina je pripravljena za uporabo: naložimo jo preprosto z LOAD "" CODE, pred tem pa natipkamo CLEAR 64255.

Rutino uporabljamo skupaj z naslednjim podprogramom v basicu:

```
999Ø LET XX = (256-8-XS-LEN D$)/2
9995 LET A = 233Ø6  POKE A, XX
  POKE A + 1, YY  POKE A+2 XS
  POKE A+3 YS   POKE A+4 8
LET A=A+4  FOR I=1 TO LEN D$
  POKE A+I, CODE D$(I)  NEXT I
POKE A+I,255  RANDOMIZE USR
64256   RETURN
```

V glavnem programu moramo pred klicem podprograma določiti vrednosti naslednjim spremenljivkam:

XY širina črk (v poljih)
YS višina črk (v poljih)
YY koordinata y vrhnje točke prve črke (računano od vrha ekrana)
XX koordinata x prve črke (računamo od leve proti desni)
D$ niz znakov, ki ga želimo izpisati povečanega.

Podprogram nam ponuja dve možnosti za izpis. Če določimo vrednost spremenljivke XX v programu in podprogram kličemo z GOSUB 9995, se izpis začne pri ▶

moj MIKRO
Titova 35
61000 Ljubljana

Izkušnje VIZ Titovo Velenje

Ob tem, ko sem prebiral revijo MOJ MIKRO in priporočila, ki so jo spremljala, sem na kratko pregledal, kaj smo v zvezi s približevanjem računalnika našim učencem naredili na osnovnih šolah v občini Velenje. Ni ravno veliko – toda startali smo!

– V prvi fazi smo spremljali informacije in obravnavali predloge, smernice in vzpodbude za uvajanje računalništva v osnovno šolo.

V tem obdobju smo spremljali tudi ustanovitev in uspešno delo dveh oddelkov računalniške usmeritve v Centru srednjih šol v Titovem Velenju. Nekateri naši učitelji in strokovni delavci so v tem času že tudi prerasli fazo igranja (na privatnih) osebnih računalnikih in začeli sami sestavljati programe za obravnavo posameznih učnih tem ali za izračunavanje statističnih pokazateljev v okviru analiz učnega uspeha.

– Vključili smo se tudi (kot mentorji) v širše podprto akcijo Mladi raziskovalci – za razvoj Velenja.

– Ob sodelovanju s Centrom srednjih šol smo vsem zainteresiranim pedagoškim delavcem omogočili informativni obisk računalniške učilnice na Centru srednjih šol.

– Ob tem smo zbrali že toliko informacij, da smo začeli nekoliko bolj kritično ocenjevati vsebino seminarjev, ki so stekli pod zvenečimi naslovi.

– Bolj smo se navdušili za seminar za naše učitelje na Centru srednjih šol – sodelovali smo pri sestavi seminarja. Ta seminar (dvanajsturni) je obiskovalo 46 naših učiteljev in strokovnih delavcev.

– Če možnosti in poti nabave računalnikov ne bi krojile tako čudne odločitve in predpisi, potem bi ti učitelji sedaj že vodili računalniške krožke na vseh naših šolah. Poleg učencev – začetnikov bi v teh krožkih delali učenci, ki so se na lastnih računalnikih že veliko naučili.

– Nove možnosti iščemo sedaj v okviru Centra srednjih šol, ki bo pod strokovnim vodstvom Inštituta Jožef Stefan iz Ljubljane organiziral računalniške krožke za učence. Zdaj zbiramo prijave učencev, v kratkem bo steklo usposabljanje učiteljev za vodenje teh krožkov in če se ne bo še kaj dodatno zapletlo v zvezi z nabavo računalnikov, potem bo delo lahko steklo.

– Krog učencev, ki bodo začeli načrtno spoznavati delo z računalnikom, se širi, zato pozdravljamo misel, da bi v reviji MOJ MIKRO objavljali njim namenjene članke.

– Revijo MOJ MIKRO bomo predstavili, skupaj pregledali in priporočili tudi na strokovnem kolegiju ravnateljev osnovnih šol.

– Seveda pa imamo že sedaj nekaj predlogov:

1. da bi v člankih usmerjali in vzpodbujali ustvarjanje programov, ki bi obravnavali učne vsebine čim večjega števila predmetov naših šol;

2. da bi objavljali prevode predstavitev tujih programov za učence osnovnih šol;

3. da bi strnili voljo, znanje in sredstva za razvoj programskih paketov za naš vzgojno-izobraževalni sistem in tako skrbeli (skupaj z Zavodom SRS za šolstvo in šolami, ki usposabljajo bodoče učitelje) za sistematično in organizirano izobraževanje učiteljev na področju računalništva;

4. da bi s čim bolj kompletnimi in zanesljivimi informacijami in nasveti pomagali šolam pri odločitvah za nabavo potrebne opreme;

5. da bi ob skrbi za kvaliteto in uporabnost vsebine revije namenili pozornost tudi oblikovanju dostopne cene.

Ob koncu pa vam sporočam še ime mentorja, ki zelo zagnano dela na področju uvajanja računalništva v osnovno šolo in ki je ustvarjalno sodeloval pri vseh prej opisanih akcijah: to je Franci Rogan, profesor pedagogike, ki dela kot šolski pedagog na osnovni šoli Anton Aškerc v Titovem Velenju, Jenkova 2.

Z zaupanjem, da nam bo revija MOJ MIKRO prinesla vedno veliko zanimivega, vas tovariško pozdravljamo!

Pomočnik direktorja VIZ
za pedagoške zadeve
prof. Dušan Dellner

moj MIKRO
Titova 35
61000 Ljubljana

XX Kadar uporabimo klic GOSUB 9990 računalnik izpiše tekst simetrično na sredino zasiona.
Povečani niz D▪ ne sme biti daljši od vrstice, ker se drugače ovije okoli zasiona.

●

Dragi bralci, verjetno »že« ima te igro Jetman, igro pri kateri je najprej se lako zagrizen igralec igre Jetpac izgubljal živce. Tipka za levo, desno, gor in laser sta najbrž lahko ugotovili. Nekaj več težav je bilo s tipko za pobiranje. To sta tipki Z ali SYMBOL SHIFT.
Najprej ja treba naložiti na vozilo bombo, potem pa še ti vanj. To naredila s tipko CAPS SHIFT ali BREAK SPACE. Ko boste v vozilu, poglejte smer sovražnikove baze. Nato se s tipko, s katero upravljate vesoljca, odpravite tja.
Če je pred vami lukna, izsiopite (z isto tipko kot za vsiop), prinesite tipko za pobiranje in nad luknjo ponovno. Luknja bo zakrpana in nadaljevali boste pot.
Tudi čas je omajan. Ko ga zmanjka vam »prijazni« sovražniki pošljejo rakeio, ki je z laserjem neuničljiva. Edino, kar vam preostane, je izsiopiti in se zaleteti v rakeio. Eno življenje gor ali dol – glavno je, da ostane vozilo nepoškodovano. Ko boste prišli k bazi, izsiopite, vzemite spet bombo in jo vrzite na rakeino izstrelišče.
Nekaterim še vedno ne bo uspelo. Tedaj poskusite s poki! Meni se je posrečilo najti še kar zanimivo »formulo«. Naložila del v basicu sliko, dolgi del in dva kratka dela. Potem prekinila nalaganje in napišila

FOR p = 40088 To 40169 STEP 3: POKE p, O: NEXT p: CONT

Naložite še zadnji del. Ko boste igrali, boste opazili veliko spremembo. »Vesoljska kolesa« skrivnostno izginejo, negativna postaja pa je, da ne vidite časa. Veliko sreče!

Rok Vrbica,
Trboje 29
Kranj

> V vaši oktobrski reviji MOJ MIKRO smo opazili napako.
> V uvodniku je namreč zapisano, da v Jugoslavijo ni mogoče uvoziti novega Sinclairovega računalnika QL z razlogom, da je v njem procesor MC 68008.
> Na srečo ta trditev ne drži, ker se sam procesor v Jugoslaviji lahko uvaža in ga naša elektronska industrija tudi že uporablja.
> Res je, da se za te komponente zahteva E U C (end use certificat), ki našega kupca obvezuje, da bo te komponente uporabljena v Jugoslaviji in je kot takšne ne bodo izvozili naprej.
> Kot zastopniki Motorola Sam do danes še nismo naleteli na problem, da naša industrija ni dobila dovoljenja od ameriške vladne administracije na osnovi E U C za uvoz katerekoli komponante visoke in najmodernejše tehnologije.
> Ne vemo, če embargo velja za Sinclairov QL kot celoto, gotovo pa to ni zaradi procesorja MC 68008, ki je v njem.
> **ELEKTROTEHNA**
> Zastopstvo Motorola Sem,
> Marjan Juvan

Pridružujem se večini vaših bralcev v oceni, da je Moj mikro revija ob pravem času in na pravem mestu. Mladim mikroračunalnikarjem in ljudi ostalim nudi celovit pregled nad dogajanji doma in v svetu. Pri pregledu oktobrske številke revije sem opazil nekaj pomanjkljivosti, ki se po mojem mnenju ne bi smele pojavljati v reviji s tega področja. Pripominjam, da sem mogoče tudi kaj spregledal, vendar bom vseeno opisal svoja opažanja.

1. Ekskluzivni test ORIC ATMOS, stran 9

V tekstu je pranos podatkov na diskalni enoti 150 K na sekundo, v tehničnih podatkih pa 250 bitov na sekundo.

2. Mimo zasiona, stran 18
V drugem stolpcu je govora o Drason Data, zdi se mi, da bi to naj bi Dragon Data.

3. Tabela programov za Sinclairove računalnike, stran 26
Manjkajo zvezdice pri programi.

4. Sola programiranja v strojnem jeziku (3), stran 38
Omenjena je tabela 3/8, ki pa je ne najdem.

5. Prva črte z računalnikom (2), stran 41

V programu 12 sem opazil ukaza GOTO 1100 in GOSUB 1200, program pa se konča z vrstico 1070.
Opomba so dobronamerna.
Želim vam še mnogo uspeha pri oblikovanju revije in mogoče več poudarka vzgoji in izobraževanju v osnovnih šolah, kar je poudaril že tov. Novinšek v prispevku 'Mikroračunalniki v osnovni šoli'.

Igor Karče,
Mozirje 114

Zdravo Mikraši!
Zelo sem bil vesel, ko sem v kiosku zagledal prvo številko Mojega mikra. Moje veselje se je le še stopnjevalo, ko sem jo prelistal. Zares vam je uspelo narediti dober računalniški časopis.

Pisem vam zaradi igre Sabre Wulf. Sploh mi ni bila lažka, kaj ti v dveh dneh sem s pomočjo vaših pokov (MM september) našel izhod. Moj naj rezultat je tak: našel sem vse štiri dele medaljona, nabral 74 240 točk in naredil 88 odstokov igre.

Ko pridete v votlino, ki jo vidite na začetku igre, vas čaka domorodec z masko. Zravan njega je vhod v še eno sobo. Če igrate s poki, sta nevarnljivi in vam do končna faza ne bo težko priti. Če igrate brez pokov, morate najti vse medaljon, drugače ne boste končali igre. Na koncu se vam izpiše reklama za naslednjo igro Underwurlde.

Naj vam dam nekaj nasvetov! Pobirajte vse, kar morate kajti to vam bo odličen kažipot. Izogibajte se volka. Če hočete priti do zgornjega gorovja, vam bo najlaže, če se ravnate po hišicah. (Hišica ni le ena.) Pazite na vijoličaste rastline.

Bralce bi prosil, če bi mi pomagali z nasveti za igri Strangeloop in Avalon.

Miran Starič,
Maistrova 12
Novo mesto

Za bralce, ki še vedno niso prišli do konca Sabre Wulfa, objavljamo še en recept za neumrtnost, povzet po oktobrski številki zahodnonemškega časopisa Computer Kontakt.
1. Natipkajte MERGE" in poženite trak.
2. Ustavite trak, takoj ko se pokaže napis OK (po nalaganju basica).
3. Vpišite POKE 23756,1 in ENTER. (Številka vrstice 0 se zdaj spremeni v 1.)
4. Popravite programsko vrstico. S kurzorjem pojdite na dvopičje po prvem ukazu LOAD" CODE. Na to mesto vpišite POKE 43575,0 (dvopičje ostane).
5. Naložite nadaljevanje programa z RUN in znova vključite kasetnik.

Vaša revija mi je zelo všeč. Žal pa moram omeniti, da ima zelo mnogo napake v člankih. Npr. v oktobrskem Mikru piše v članku o računalništvu v bankah, da ima Ljubljanska banka IBM 3031 in IBM 3158, vsakega s po 4 megabiti! To bi bilo »prelopljeno« v byte 500 Kbytov!
Na strani 18 pri nas imajo nekateri spectrume 80 K!
Zaradi tega lahko dobimo bralci napačne predstave o računalniških komponentah.
Všeč mi je tudi vpliv Mikra na revijo Bit, saj je-ta ni več tako monotona (verjetno zaradi borba za bralce).
Želim vam še veliko uspeha pri urejanju revije!

Janko Kogelnik,
Breznica 19 a
Pravalja

> ## BANTKONTRA
> *Mahkovojni redakciji Radija Student*
> Ko grem VEN iz križišča, zavijem v Savinjsko dolino in tam vzamem hmelj, ki ga odnesem v Laško. Tam jim kot Ribničanom primanjkuje surovin in mi zato zamenjajo HMELJ PIVO! To pivo potrebuje neki pijanec v Novem Cityju, ki pa mi da za PIVO SLIKA. Nato grem v Črnigrad in tam s pomočjo DOLAR-ja in GOR vzamem potni list (zaradi slike je veljaven). Potem se mimo Jerneja prebijem VEN. Naslednja postaja je Kolinska. Tam se oskbim s krofom. Potem grem na Zale CIRA CARAT! Klapcu dam krof in vzamem dile. Potem spet na Žalah sedam in grem v Rejbenco. Daske Sod pa imam sod z OD. Nato emigriram in poberem die Flinte und die Porno revue und der Joystick. Naslednja faza je kviz, na katerem dobim Atašo. Na ... vzamem obleko, skočim še v Bombazcviran po kravato, vse skupaj oblečem in že sem dober za Vido. Pred njo pa še dam Rožletu porniče in Kakcu piščalko. Kakec mi pove, da Bedancu s pusko spizdim. Pod Peco je Matjaz z umazano brado in mi za Attacha prepusti meč. Za potrabno Varoniko mi da Friderik lonec z denarjem alias čekovni blok. Zakaj ti v Goranju ne dajo zanjo Televizorja? Potem grem zopet na lepš v Vzhodno dazalo. Tudi tu nekaj ne štima. S kravato hočeš ili v disco pa ti napiše: Plavam p. p. p. kot zamorc –
> Z rokavicami, ki jih dobim pri vas, naberem Volcin in z njim polešim Ajdovsko deklico. Ta mi sicer da moko, toda pekarna v Puli ne dela (mogoče jo prenavljajo in bo muzej našega boksa?). Potem pride na vrsto zadnja finta v Iskri: pomagam, cakam in dalam (ali pri obrelno) in dobim tastaturo za HR-1884 pr. n. š. To trampim va Zagrebu za Turbo-usisivac, s katerim pa ne vem, na kod na kam.
> THE END
> P.S.!
> Zakaj lenoba v sv. četverki noče delat? Kar povejte mu, da mu jaz že ne bom kobilice nosil!!! In še to, zakaj sta bili tako nesramni in si izmislili čarobno besedo pri Figovcu. Plezil sem sa pod kioski pa ni bilo ničesar!? **Zahtevam, da se mi v Mojem mikru javno opravičite!**
> **Marko Kušar,**
> Dolenjska c. 110
> Ljubljana

Skrivnost Diabola

MIHA REMEC

Belo sonce je zahajalo, izza nazobčanega obzorja na vzhodu se je vzdigovala velikanska obla temno rdečega sonca Dan na planetu Diabolu se je prevešal v rdečo noč Prilikavo sluzaslo rastlinje podobno vodnim algam se je svetlikalo v spreminjasti svetlobi Bilo je mokro saj je na Diabolu kar naprej rosil lepljiv po amoniaku smrdeč dež.

Jeremija Rak vodnik rudniškega robota je nestrpno pogledoval na časomer njegova izmena je bila pri koncu v glavi je čutil skeleče sršenje pred očmi se mu je meglilo in na vso moč se je moral zbrati da je na zaslonu vodil gibe mehanične pošasti ki je globoko pod površjem v razmerah pogubnih za vse živo kopala in nakladala rudo

To delo ga je ubijalo Ni vedel kaj kopije ruda na zaslonu je bila videti zelena barve z rumenimi žilami in naseljenci na Diabolu so ji rekli kar ruda Še nekaj je glodalo Jeremijo večkrat se je vpraševal ali zares rudari ali pa je tudi njegovo vsakdanje opravilo samo ena izmed iger na zaslonu Do rudišča ni bilo dostopa in tako se ni mogel prepričati ali roboti spodaj res obstajajo

Zamenio je pričakal z vzhidom olajšanja

»Kaj posebnega?« je vprašal David ko je zasedel prostor pred zaslonom in prevzel robota

»Pospešujemo nakladanje Tovorni premičnik se jutri vrača na Zemljo « je odgovoril Jeremija in se prijel za senca »V glavi me skali «

»Zunaj je spet ionski vihar « je precedil skozi zobe novi vodnik »K sreči nisam dovzeten Pojdi v zabavišče in pozabi «

»Najbrž bom res šel Ne bom ti želel lahka noči David «

»Hvala zelo obziren si « sa je trpko nasmehnil mladenič in zadnji hip umaknil robota pred nevarnim zruškom

Jeremija je naglo zapustil vodilnico, kjer sa je truma vodnikov za zasloni ubadala z mehaničnimi rudarji V hodniku ki je zavijal v naselbino se je moral ustaviti Skelenje v glavi je postalo tako hudo da se ja oprijel stene in začel s čelom butati ob gladko kristalovino okenske line Zunaj je v zamolklo rdeči noči sevala bela svetloba in se pahljačasto razpirala čez obok neba

Bolečina je prešla nenadoma in Jeremija je spet zadihal Prehodil je hodnik in stopil v naselbino Vsa je bila pokrita s prozornimi oboki ki so varovali naseljence pred nadležnim smrdljivim dežjem in surovim ozračjem Diabola mraza v rdečih nočeh ko je čez nebes polzela Kerma in peklenske dnevna vročine ko sa je pod belikavimi žarki Alba ozračje spremenilo v soparen kotel

Pri prehranjevalnem avtomatu se je okrepčal in pohitel v zabaviščne prostore, čeprav ga ni mikala zabava Ni mogel razumeti naseljencev ki so iz zavarovanih lovskih prez z ostrostreli pobijali jauhe bele sluzaste pokveke ša najbolj podobne velikanskim žabam Zdolgočaseni okrutnezi so jih namenoma sebe zadevali in se zabavali ob božjastnem poplesavanju in trzanjem krakov smrtno ranjenih živali Tudi igralcev v računalniški igralnici ni razumel saj so bile vse igre podobne vsakdanji zaposlitvi v rudniški vodilnici enaki zasloni enake tipke enaki ukazi. Nili ga ni mikalo razgibavališče čeprav bi tam lahko plaval in igral diabolo pradavno igro z napeto vrvico in ulorjenim vretenom Nikoli ni zvedel ali so planet Diabolo poimenovali po tej igri ali diaboličnosti slutil pa je de se ludi za tem skriva neka hudičeva igra

Hitel je drugam Nezadržno ga je vleklo k Evi

Eva ja bila ženska z druge strani zaslona v oddelku za čutne igre Jeremija ji je dal ime Za vse druge je bila le ženska iz celica 6969

V oddelku za čutna igre je stala vrsta celic jočih na ploščice s katerimi si lahko vklopil ogledovalni zaslon Vedno znova se je počutil ponižanega in osramočenega kadar se je moral postaviti med moška ki so si s prostaškimi pogovori krajšali čas Peklilo ga je trpko spoznanje da lahko vsak od teh pohotnežev zahteva ploščico 6969 in si pridobi polurno pravico do gledanja Eve In dobro je vedel kaj pomeni to ogledovanje

Na Diabolu ni bilo žensk Ko so med trumami brezposelnih Zemljanov novačili naseljence za Diabolo so jih opozorili da tam ne bo žensk sevanja ionski viharji in surovo ozračje vse to bi naj pogubno vplivalo na žensko rodnost in duševnost Otroci spočeti na Diabolu bi se najbrž rodili kot pokveke Zato so prepustnico za ta svet dobili le moški oddelek za čutne igre pa jim je ponujal nadomestka ženske lutke in ogledovalne zaslone

Naposled je Jeremija prišel do razdeljevalca ploščic Številka 6969 je bila prosta in z radostjo v srcu je pohitel po dolgem prehodu do celice

Znova in znova ga je ta prostor spomnil na stranišče Belo obložene stene odplakovalna tla lijak papirnata brisača Kljub močnemu razsmrajevalcu se je v celico zajedel vonj po znoju in slabo umitih moških

S tresočo se roko je potisnil ploščico v vklopno režo Takoj se je zbistril zaslon ki je obsegal ves čelni zid V ozki celici na drugi strani zaslona se je prikazala Eva Oblečena je bila v oprijeto svetlikajočo se odelev ki je izzivalno poudarjala njene obline imela je razkošne grudi dolge črne lase čutne ustnice in jamice v licih Lesket njenih temnih oči je Jeremijo prevzel kot hipnotično iskrenje Vsakokrat ko se je srečal z njenim pogledom je bil zbegan in očaran

Kakšna tehnika je to da ustvarja tako popolno podobo? Nobena druga globinska slika se ne more primerjati z njo Imel je občutek da dekle zares stoji na drugi strani Pri vsej tej slikovni popolnosti pa ni bilo zvočne ga slike Zakaj?

»Eva!« je poskušal z ustnicami oblikovati besede »Kje si?«

Skomizgnila je z rameni in se mu nasmehnila Mar ne razume vsezemeljskega jezika?

Nepremično ga je gledala v oči s počasno kretnjo povlekla čez rame stransko razpiravko oblake in mu izzivajoče pokazala dojko

»Povej mi kje si!« je vztrajal »Si tu?« Pokazal je na njeno stojišče

Spet ga ni razumela Polegnila je razpiravko ob bokih do narta in se počasi izluščila iz oblačila Val razburjene krvi je preplavil njegove žile V glavi mu je bučalo in šumelo Strmel je v najlepše golo dekle kar jih je kdaj videl V ihti nemoči je začel s pestmi razbijati po zaslonu

»Eva, rad te imam!« je kričal »Žalim te!«

Zdaj ga je menda razumela Oči so se ji razprle in orosile kot mulavka je segla k srcu in mu ga s kretnjo navidezno ponudila Potem se je tudi ona stisnila k zaslonu osuplo je gledal kako se ji dojka sploščajo ob dotiku s kristalovino videl je ustnice ki so se prilepila na prozorno ploskev To ne more biti slikovni program to ni igra in če je potem se nima nihče pravice igrati s tem

Jeza in neznosna žalja po zaresnem stiku z Evo sta mu dali nadnaravsko moč odtrgal je težal lijak in ga treščil v zaslon

Kristalovina se je žvenketaje sesula v tisočero iskrečih se delcev

Jeremija Rak je ostrmel podoba Eve se ni zdrobila Dekle je še vedno stalo pred njim, živo osuplo in prestrašeno

Planil je k njej in se začel kot slapec dotikati njenega telesa Prepričati se je hotel da Eve ni prikazen na zaslonu ampak resnično otipljivo bitje Ko je to ugotovil se je je oklenil kot utopljenec rešilne bilke

»Torej si tu « je jecljala Eva »To je vendar ženski planet tu ni moških «

Jeremijo je popadel krčavit nepremagljiv smeh

»Nam so pa rekli, da je Diabolo moški planet «

Ni je več izpustil iz objema Poiskal je njene ustnice in se hlastno zasesal vanje

»Ne smeva « se je zbala

»Smeva « je odločno rekel Jeremija »To je zdaj najin planet «

In potem sta se združila z dolgo zadrževano željo in hrepenečima telesoma

●

V višjem nadzorništvu planeta Diabola se je na zaslonu omrežja naselbina na točki 6969 prižgala opozorilna luč in predirno zaporedno piskanje je naznanjalo da je diabolski igralni sistem v razsulu

Čudoviti svet dodatkov: Commodore

CIRIL KRAŠEVEC

Naj ne bo že na začetku pregrobe ocene (in prehud udarec) da je v splošnem za Commodorove računalnike manj raznovrstnih hardwarskih dodatkov kot za njihove črne konkurente. Trditev kaj hitro zdrzi, saj je treba poudariti, da je po hardwarski plati commodore precej bolj popoln računalnik od spectruma in tako tudi industriji, ki živi od Cliveove skoposti, ni odprto toliko manevrskega prostora.

Zakaj je commodore bolj popoln? Pojdimo po vrsti! Ima profesionalno tipkovnico, standardne priključke za igralne palice in na ravno idealen, ampak vseeno bolj pripraven izhod za povezavo z zunanjim svetom kot spectrum. Če pogledamo malo po prospektu, kaj še dela proizvajalec, vidimo, da lahko dobimo vse, kar potrebujemo. Pogon za gibki disk, kasetofon, tiskalnik in risalnik lahko priključimo na računalnik, ne da bi potrebovali en sam vmesnik.

VIC-1530C2N datassette

Ta kasetofon malce nenavadne oblike in s posebnim kablom je posebej primeren za delo z računalniki tipa commodore. Zakaj posebej primeren? Morda zato

ker brez spajkalnika ne moremo priključiti običajnega kasetofona, ali zato, ker je zapis na trak s posebnim kasetofonom boljši, zanesljivejši. Pri polževih hitrostih prenosa bi bilo morda prav vseeno, kakšen je kasetofon, saj ni tako imenovani datassette prav nič drugačen od kateregakoli standardnega kasetofona. Manjka mu samo nizkofrekvenčni avdio del, za zameno pa ima nekaj malega elektronike (beri MM št. 1). Proizvajalec se je s posebnostmi rešil vseh mogočih domačih ropotij, s katerimi imajo uporabniki največkrat samo težave, in zaslužil 100 mark pri kosu. Morda je smešno le to, da se je odločil za poseben kasetofon in ostal pri tako počasnem prenosu podatkov (nekje okrog 200 baudov). Najbrž je boljše vrabec v roki kot golob na strehi!

VIC—1541 single-density disk drive

To je že boljša kombinacija. Standardni 5,25-inčni gibki diski, na katere lahko spravimo 170 K za

dodatno ceno okoli 800 nemških mark. Prenos ni ravno običajno hiter, saj po serijski povezavi tečejo podatki s hitrostjo samo nekaj več kot 3000 baudov. Zmogljivost diskov je dvakrat večja kot pri Atarijevih (88 K) in nekaj večja kot pri Applovih računalnikih (114 K). Razlika med starim pogonom z oznako 1540 in tem je samo v novem romu. Diske, posnete na starem pogonu ali na pogonu CBM 4040 (dual disk drive), lahko beremo tudi s pogonom 1541.

Na serijska vrata lahko priključimo več kot štiri pogone. Krmilnik diska (mikroprocesor 6502) in disketni operacijski sistem sta vdelana v samem ohišju. Zaradi takšne izvedbe je mogoče pisati programe, ki se izvajajo že v samem diskovnem pogonu. Zanimiva možnost. Vendar se to pokaže za nepraktično, kadar nastane napaka. Disk se vrti, lučka utripa, računalnik pa čaka in ne ve, kaj je narobe. Za primer lahko poskusite posneti kaj na poln disk. Disk se bo vrtel, lučka bo utripala, mi pa bomo zvedeli za napako šele takrat, ko bomo vprašali računalnik.

Programi v basicu se enostavno duplicirajo tako, da jih vpišemo v pomnilnik in posnamemo na nov disk. Za vse druge programe boste potrebovali dva disketna pogona ali zvrhano mero znanja.

VIC-1525 dot matrix grafični tiskalnik

Več o tem tiskalniku smo pisali v tretji številki naše revije. Povejmo samo to, da tiskalnik lahko piše ves Commodorov nabor znakov. Znaki, kot so j, p, y ali g, niso znižani, tako da lahko večkrat preberemo slavna namesto glavna vrata. Hitrost pisanja je 30 znakov na sekundo. VIC-1525 se na računalnik priključuje na serijska vrata ali na zadnji disketni pogon, priključen v verigo.

Drugi Commodorovi tiskalniki se priključujejo na standardni vmesnik IEEE-488. So pa tudi vmesniki, ki omogočajo priključitev Epsonovih ali katerih drugih tiskalnikov s paralelnimi vmesniki centronics. Za vmesnik je potreben še krajši program, ki bo Commodorove znake pretvoril v standardne znake ASCII.

VIC-1520 plotter/printer

Pravi risalnik za commodore. Štiri barve, malo ožji papir in vse skupaj na pult za okoli 300 nemških mark. Risalnik/tiskalnik (plotter/printer) smo v naši reviji tudi že predstavili, samo da je takrat ravno take izdelke zapiral v svoja ohišja Sharp. Na področju računalniške periferije se je tako kot pri kasetofonih pojavil izdelovalec mehanike (seveda japonski), ki oskrbuje s svojimi izdelki vse mogoče firme. Vse, kar naredijo »pravi« izdelovalci naredijo, je to, da risalnik pritrdijo v svoje ohišje in

ga povežejo po svojih standardih. Tak je tudi Commodorov risalnik s peresom (ball point). Pri risanju premika papir naprej in nazaj, pero pa levo in desno. Piše lahko 80, 40, 20 ali 10 znakov v vrstico, in to s hitrostjo 14 znakov na sekundo.

VIC-monitor

Tovarna Commodore je že od nekdaj znana po tem, da izdeluje skoraj vse. Tako se je v njihovem proizvodnem programu znašel tudi barvni RGB monitor. Priključimo ga na ustrezen izhod računalnika. Monitor meri v diagonali 14 inč. Cena je nekaj manjša od 1000

nemških merk. So pa tudi tosforjevi zeleni monitorji z diagonalo 12 inč. Stanejo okoli 300 merk.

VIC-1600 modem

Za komunikacijo med uporabniki je Commodore izdelal modem, ki vas po telefonu poveže z banko podatkov ali drugim računalnikom. Hitrost prenose podatkov po telefonskih vodih je 300 bitov na sekundo. Cena je v ZDA 100 dolarjev. Tako nizka je zato, ker modem uporablja za prenos standardni vmesnik RS 232 C. V ceno modema je vključena tudi enoletna naročnina na tako imenovani Compuserve, ki ga kupec sam izbere. Compuserve je ime Commodorove informacijske mreže, ki ima svoje podružnice skoraj po vsej Ameriki.

Delo z modemom ne zahteva posebnih sposobnosti. Če na primer odtipkamo GO CGM potem, ko dobimo odziv Compuserve, smo že v banki informacij. Lahko prebiramo elektronsko pošto, odgovarjamo na vprašanje velikega računalnika ali se igramo video igre. Modem lahko uporabljamo tudi za kontaktiranje s katerimkoli drugim informacijskim centrom.

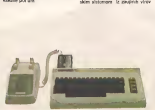
Je pa ta modem dokaj nenavaden. Pri napravah takšnega tipa smo navajeni, da telefon izključimo iz vtičnice (kamor priključimo modem) in ga priključimo na novo napravo. Prednost tega je, da lahko telefon uporabljata bodisi računalnik ali človek, ne da bi vedno znova vse pretipkavali. Modem VIC pa se priključuje na telefonsko slušalko. Najprej odvijemo mikrofon in na njegovo mesto priključimo modem. Zvezo vzpostavimo tako, da izberemo telefonsko številko, in ko slišimo znak, začnemo delati prek računalnika.

Kaj pa, če nas takoj po prekinitvi dela pokliče prijatelj? Zvezani v telefonske kable mu bomo rekli, da ga bomo poklicali nazaj čez kakšne pol ure.

CP/M v CBM-64

Takšni dodatki so vredni zlata. Poslovni uporabniki CBM-64 so dolgo čakali na izpolnitev te obljube. Res je proizvajalec pred kakšnim letom poslal na tržišče ploščico za dodatni mikroprocesor Z 80. Nič hudega, če je tak mikroprocesor tudi v spectrumu. Je pač tako, da dela standard CP/M samo s tem mikroprocesorjem in če bomo želeli uporabljati programe za vse računalnike CP/M, bomo prenesli tudi to sramoto.

Na čudežnih ploščicah, ki so se pojavile v ZR Nemčiji, pa je napaka, ki večini uporabnikov sploh ne pusti delati z novim operacijskim sistemom. Iz zaupnih virov smo zvedeli, da so ploščice umaknili iz prodaje. Moj mikro pa bo spretnim in veščim kmalu po novem letu ponudil načrte za izdelavo ploščice za operacijski sistem CP/M.



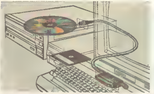
Manjši dodatki različnih proizvajalcev

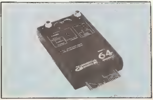
Sem sodijo razna vezja za kontrolo gospodinjskih aparatov, uporabo računalnika v merilne in krmilne namene, predvsem pa je veliko dodatkov, ki krmilijo vse mogoče železnice in avtomobilska cesta. Med takšne dodatke smo vtaknili tudi vmesnike za povezavo commodora z glasbenimi instrumenti (sintetizatorji) in video diski.

Za slabši okus v ustih in izziv domačim graditeljem poglejmo dva takšna dodatka.

Prvi je namenjen glasbenikom in mu pravijo sekvenčnik MIDI. Kratica pomeni Musical Instruments Digital Interface. Angleži pravijo, da je ta dodatek nova beseda za vaš slovar računalniškega govora.

MIDI je naprava, ki zna sintetizirati zvoke, je tako imenovana ritemska mašina in še sekvenčnik. Sintetizator (lahko jih je tudi več) povežemo z našim dodatkom po dveh vtičnicah DIN. Na teh vtični- ▶





cah so vhod, izhod in prehodna vtičnica. Med tujimi ocenjevalci te naprave so se slišali pomisleki o dovolj veliki hitrosti serijske povezave. Vendar je proizvajalec s svojo odločitvijo ustregel standardom za povezavo tovrstnih glasbenih instrumentov. MIDI komunicira z računalnikom asinhrono s hitrostjo 31.25 Kbaudov.

Povezavo z računalnikom so izvedli z Motorolino ACIA 6850. Vse potrebne podatke za programiranje boste našli v priloženih navodilih. Če si želite takšno napravo, se obrnite na naslov: MIDI Users Group, 8426 Vine Valley D R. Sun Valley, CA91352, USA.

Drugi dodatek izdeluje Digital Research in se imenuje VidLink. Namenjen je izdelavi resničnih video iger. Za to izdelavo potrebujemo poleg vmesnika laserski video z nekaj diski. Najprej vse skupaj lepo spojimo s kablom in nato začnemo programirati. Nekajkrat pregledamo videodisk in se odločimo za potek dogodkov. Namesto da bi video material gledali lepo od začetka do konca, bomo sami določili vrstni red sekvenc. Pravijo, da je res zabavno. Škoda le, da ne moremo sami snemati video diskov. Cena vmesnika VidLink je 49 dolarjev. Če imate zelence in vam bo vmesnik dobro delal, nas povabite na ogled svoje prve video kolobocije!

Postanite letalski as!

ROMAN SNOJ

Blue Max

Podjetje Synapse Software ter avtorja Bob Polin in Peter Adam (za glasbo je poskrbel Stephen C Biggs) so izdali igro-simulator Blue Max, ki sodi med najboljše dosežke za commodore 64 Grafične in glasbene zmogljivosti računalnika so izkoriščene v polni meri

Igralec se postavi v vlogo pilota dvokrilce, ki mora med I svetovno vojno povzročiti čim več škode sovražniku Vzlet Toda pozor! Vsak premik igralne palice, dokler hitrost ne naraste na 100, se tragično konče Računalnik vpiše v spodnji del zaslona tvoje letalske sposobnosti v tem primeru kamikaza Po uspešnem vzletu – hop na junaška dejanja. Na voljo imamo mitraljez in bombe Teren, nad katerim letimo je poln nasprotnikovih objektov tu je most, tam stavbe, drugje ladja cesta

Ko počistiš nasprotnike ob reki se pomakneš v notranjost kjer so letališča gospodarski objekti ceste itd Ves čas se bojuješ tudi z nasprotnikovi letali Ker pa je zaloga goriva omejena sa moraš spustiti na svoje letališče, dotočiti gorivo in popraviti poškodbe Če se ne spustiš tam, se zadeva po navadi konča z neuspešnim zasilnim pristankom ob kakem drevesu. Pomemben je tudi pravočasen pristanek da se letalo ustavi ob hangarju Sicer je ponovni vzlet enak zasilnemu pristanku, v spodnji del zaslona pa se vpišejo tvoje sposobnosti

Program je mojstrsko dodelan Vse se sliši resnično zvoki letalskih motorjev vseh vrst streljanje, tet bombe, eksplozije Po panorami terena se premikajo celo sence letal

Igra ima tri parametre, ki si jih lahko izbiramo Najprej je tu izbira težavnostne stopnje Izberemo lahko med začetnikom letalcem in ekspertom Sledi izbor kako bomo pilotirali normalno ali pilotsko Za pojasnilo pri normalnem pilotiranju moraš za dvig letala potisniti ročico navzdol Tretji parameter je vpliv gravitacije da ali ne. Za določitev vseh parametrov so funkcijske tipke S pritiskom na F1 izpiše računalnik tabelo parametrov F3 izvaja prehod iz enega parametra v drugega S tipko F5 določimo želeno stopnjo v vsakem parametru Igra se začne s pritiskom na F7 ali gumb na igralni palici

Zaslon je pri igri razdeljen na dva dela Večji del je namenjen panorami terena nad katerim letimo v ozkem spodnjem delu pa so kontrolni izpisi Ti se nanašajo na hitrost, višino količino goriva in streliva, vrste poškodb, oznake pomembnejših objektov itd Letalo ima lahko največ štiri vrste poškodb, ker ustreza štirim zadetkom protiletalske obrambe streljanje in metanje bomb odpovesta, nemogoče je manevrirati v navpični in vodoravni smeri Vsak peti zadetek je smrtonosen in moraš igro spet začeti s pritiskom na F7

Ni pa nujno, da te zadenejo samo v zraku Ko se spustiš na svoje letališče in dotekaš gorivo, se nasprotnikova letala »sprehajajo« nad tabo in meni se je že zgodilo da so me zadela z bombo V kontrolnem izpisu se pokažejo tudi črke ki označujejo pomembnejše objekte L pomeni domače letališče, W je zelo pomemben objekt P je nasprotnikovo letalo zadetek pa prinese dvojno število točk Ob strani spodnjega dela zaslona so puščice ki sporočajo, ali je letalo više ali niže od našega Kadar sta letali v isti višini se ves spodnji del obarva svetlo modro Takrat hitro pritisnite na sprožilec in uspeh ne bo izostal

Zaxxon

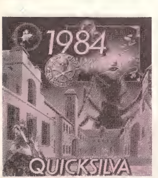
Peter Adam, za spremembo sem in za nemško podjetje GCS nadaljuje z Zaxxonom svojo serijo dobrih simulacijskih iger Tokrat se je obrnil v prihodnost in nastala je igrica, ki v ničemer ne zaostaja za Maxom

Kot pilot bojnega letala morate premagati tri prostore in na koncu uničiti robota, za nagrado pa dobite igro na višji stopnji Vse se dogaja v tridimenzionalnem prostoru Prvi in tretji prostor imata obliko labirinta, v katerem so topovi, rakete cisterne z gorivom itd Čim več jih uničite tem več točk dobite Poskrbljeno je tudi za to da se ne bi kdo v začetku sprehodil skoz labirint brez boja. V tem primeru ostane v drugem prostoru brez goriva Gorivo si priskrbiš tako da uničiš čim več cistern

Ko v prvem prostoru premagaš vse ovire, tj preletiš vse odprtine se založiš z gorivom in sestreliš vse rakete, ki te ogrožajo, te čaka vesoljska bitka Nasprotnikove vesoljske ladje so razporejene po vsem tridimenzionalnem prostoru in se stalno gibljejo Tvoja naloga je da med letom skoz prostor sestreliš čim več letal V pomoč je zvočni signal, ki te opozori, kdaj je sovražna ladja na muhici

Po spopadu (četudi nisi uničil vseh letal) se odpraviš v tretji in zadnji prostor Ta je dejansko prav tak kot prvi razlika je le da prehodi niso več prosti, temveč so zaščiteni posebni žarki te pokončajo če nisi v pravi višini Zadnje dejanje opraviš pred robotom Če se ti je posrečilo je nagrada igra na višji stopnji V prvem in tretjem prostoru ti pomaga tudi višinomer

Na voljo so tri življenja ki jih lahko povečaš še za dve Igro nadaljuješ na začetku tistega polja, kjer si zgubil življenje Igrate lahko dva igralca Tudi tu se uporabljajo funkcijske tipke. F1 povečuje življenja na štiri ali pet S tipko F3 določimo način streljanja (rafali ali posamezni streli) V začetku je bolje streljati v rafalih, saj potrebuješ za sestrelitev rakete vsaj tri strele F5 določa število igralcev F7 je za začetek igre



Z bombami nad mravlje

IZTOK STRAŽAR

Ant Attack (Napadk mravelj) je ena najbolj priljubljenih iger leta 1984. V zapuščenem mestu, katerega edini prebivalci so velikanske mravlje, leži nemočna deklica (ali deček, to je odvisno od naše izbire na začetku) in kliča na pomoč. Kdo ne bi poiskal in rešil ubogega otroka?

Zanesemo se lahko le na svojo spretnost in na nekaj eksploziva, ki nam bo prišel še kako prav pri obrambi pred mravljami. Zametanje bomb so tipke od A do G, lepo po vrsti, tako da lahko z vsako zalučamo smrtonosno bombo za eno stopnjo dlje. Paziti pa moramo, da ne bombardiramo v tisto smer, kamor se premikamo. Če ste se že naveličali igre, s tem najlaže naredite samomor. Tako sicer ne boste slišali besed, kot so »Hello, my hero« (Pozdravljen, moj junak) ali kaj podobnega, umrli pa boste častno in hitro, ne da bi vas grizli mravljinci.

Če niste preveč spretni in ne verjamete, da so mravlje vztrajne živali, bo najbolje, da prepustite svojo izvoljenko (ali izvoljence) nedolžnemu spanju, sami pa jo ucvrete čimdlje iz mesta. Imeli boste kaj videti. Mravlje se bodo pognale za vami in njihov galop bodo lahko ustavile samo granate.

Ant Attack ima deset stopenj. Na vsaki od teh so štiri mesta, kjer ste lahko deklica ali deček. Izjema je prva, najlažja stopnja, kjer je le eno tako mesto.

Dogajanje na zaslonu je mogoče opazovati iz štirih zornih kotov, ki se v trenutku spremenijo, s pritiski na štiri skrajno desne tipke: BREAK, ENTER, P in O.

Večino dela se da opraviti v začetni perspektivi in gotovo vam je mesto najbolj znano prav iz tega zornega kota. Toda nekaj skritih lokacij boste odkrili le z zamenjavo perspektive.

Katere so te lokacije?

3 A: skriti otrok je delno viden tudi iz perspektive BREAK, pomagamo pa si z ENTER ali s P

4 D: P
5 C: P
6 A: O ali P
6 D: ENTER ali P
8 C: P

Najlaže je, če greste tudi do teh lokacij v tisti perspektivi, ki jo najbolje poznate, in šele na kraju sama spremenite razgled.

Na 8. stopnji je dostop do skritega otroka malce težji kot prej, kajti s tal se ne da skočiti dovolj visoko. Zato je treba počakati na mravljo, skočiti nanjo in se od nje odriniti na želeno mesto. To je treba storiti tudi na deveti in deseti stopnji.

Ubite mravlje ne povečajo rezultata, ampak ga zmanjšajo. Odvisen je namreč od časa, ki vam ostane. Medtem ko se bojujete z mravljami, čas seveda teče. Rezultat se določi tako, da se ostali čas zmnoži s številko stopnje, ki ste jo zadnjo premagali. Če vam npr. na 3. stopnji ostane 700 časovnih enot, bo novi rezultat enak prejšnjemu + 3-krat 700. Zato si prizadevajte, da boste v nenehnem gibanju (ne le okoli svoje osi). Kadar se premikate v smeri skrite deklice ali dečka, je oznaka SCAN v spodnjem desnem kotu zaslona zelene barve, drugače je rdeča.

Kjer so skrivne lokacije blizu obzidja, lehko privarčujete nekaj časa: mestne ruševine zapustite s skokom čez obzidje. (Tokrat je to težje, ker morete hkrati skočiti na mravljo oba, deklica in deček.)

Med iskanjem drzite desni mezinec na tipki BREAK. Slike na zaslonu se bo spremenila hitreje in imeli boste boljši pregled nad dogajanjem.

Mojstrski rezultat je več kot 45 000.

Kje sta deklica in deček?

Lokacija, kjer se lahko skrivata deklica ali deček, so označene s številko od 1 do 10 (stopnja) in črko od A do D (štiri črke – štiri različna mesta na isti stopnji).

Oznaka z zvezdico pomenijo, da so skrita mesta vidna iz drugečne perspektive kot BREAK. St = stopnja, kv = kvadrat na zemljevidu.

Na prvi stopnji je skrita lokacija v kvadratu 2 IV.

st	kv	st	kv	st	kv
2 A	1 IV	3 A*	2 I	4 A	1 IV
2 B	3 IV	3 B	2 II	4 B	3 IV
2 C	3 III	3 C	3 II	4 C	4 IV
2 D	3 III	3 D	4 IV	4 D*	3 IV
5 A	1 II	6 A*	3 I	7 A	3 II
5 B	4 II	6 B	2 I	7 B	3 I
5 C	1 II	6 C	1 I	7 C	4 I
5 D	4 II	6 D*	3 I	7 D	1 II
8 A	1 I	9 A	1 IV	10 A	1 III
8 B	2 I	9 B	2 I	10 B	2 III
8 C	4 I	9 C	1 IV	10 C	4 IV
8 D	4 II	9 D	4 I	10 D	4 I

Prvih deset
Mojega mikra
(4.) 1. Match Point Psion spectrum 48 K
(3.) 2. Pinball Sagittarian spectrum 16 K
(–) 3. Full Throttle Micromega spectrum 48 K
(1.) 4. Jet Set Willy Software Projects spectrum 48 K
(8.) 5. Soccer Commodore CBM-64
(10.) 6. Sabre Wulf Ultimate spectrum 48 K
(–) 7. Football Cup Artic spectrum 48 K
(7.) 8. Mugsy Melbourne House spectrum 48 K
(6.) 9. Atic Atac Ultimate spectrum 48 K
(5.) 10. Scuba Dive Durell Software spectrum 48 K
GOSUB

Zmagovalec nenavadnega žrebanja je:

MIRAN

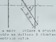
Rešitev ni bilo toliko, kot "zrn", toda vse prispele rešitve so pravilne. Tudi brez računalnika je jasno, da je zmagovalec Miran, torej fant, ki dobiva točke kadar 5cm dolga igla seka med seboj 3 cm oddaljene črte. Še dobro, da je bila Miranova zmaga tako jasna. Dobili smo zelo različne rešitve, od popolnega Jankovega poraza s štirimi odstotki, ki jo je poslal Matjaž iz Šentvida pri Ljubljani, do skoraj neodločenega izida, ki ga napoveduje Renata iz Ptuja.

Še ena statistična značilnost. Razen Renate je pravilno rešitev poslalo le eno dekle. Morda so se naše reševalke ustrašile kotnih funkcij, ali pa preprosto raje poslušajo slovenščino.

Objavljamo rešitev, ki je kratka in jasna:

Z veseljem sem prebral vseh pet vaših revij. ... [illegible] Tudi nalogo z zadnjih strani sem že reševal, vendar [illegible] ... Problem sem si ... [illegible] ... ne preveč zahteven ... [illegible]

Tu pa je navedena moja rešitev



[illegible] ... oklepa s pravokotnico na črte kot ALFA in je njeno središče od dolžine X oddaljeno od najbližje črte. Pri tem veljajo [illegible]

```
1   0 <= ALFA <= PI/2
2   0 <  X    <= 1 5
```

Pogoj, ki more biti izpolnjen, če naj šivanka ne seka črt, je

```
X > 2 5 * COS (ALFA)
```

T[illegible] je hitro [illegible] Jankove šanse. Miranove zmage lahko upoštevamo kot Jankov poraz ... [illegible] informacije o Jankovem dobičku dovolj, da izvemo za izid igre. Če je ta večji od 0 ... [illegible] Janko zmaguje, če manjši, izgublja. Ko je absolutni dobiček dosegel neko dovolj veliko vrednost, lahko iz predznaka sklepamo, kdo od ... fantov je skupni zmagovalec.

Program ... [illegible] ... ima to zanimivo lastnost, da se spretnost ... vnaša v računalnik kot se naloži s kasete.

```
10 LET d=0
20 LET d=d + 2*(RND*1 5 > 2 5*COS(RND*PI/2)) - 1
30 PRINT AT 10,0,"JANKOV DOBIČEK   " d "   "
40 IF ABS d < 100 THEN GO TO 20
```

Po programu ... [illegible] se na vsakem 'metu šivanke' izpiše JANKOV DOBIČEK. Lahko opazujemo, kako se dokaj hitro in vztrajno odda... Na koncu doseže vrednost -100. Upravičeno lahko sklepamo, da bo igro dobil Miran, torej tisti, za katerega zmaga ... [illegible], da šivanka črte seka.

Pozdrav
Bojan Cestnik
Pol 67
61353 Borovnica

Dve rešitvi sta tudi krajši, dolgi samo eno vrstico. Avtorja sta nalogo rešila analitično, računalniku pa dovolila, da izpiše rezultat. Še trije lastniki ZX Spectrumov so nalogo rešili v štirih vrsticah, lastnik CBM 64 pa je za identično rešitev moral uporabiti vrstico več. Kaseto za ZX Spectrum je žreb prisodil

Bojanu Županu,
Smarca 39c
61240 Kamnik.

Po 500 ND pa dobijo:

Bogdan Filipič, Črtomirova 12, Ljubljana
Janko Uhan, Vrečkova 11, 64000 Kranj
Tomaž Martinčič, Vipavska 30, Ljubljana
Franc Mogan, Šalek 10, 63320 Titovo Velenje
Viktor Kraševec, Ulica Koroškega bataljona 7, 61231 Črnuče
Davor Gornik, Kosarjeva 35, 62000 Maribor
Rudi Kofol, Rižana 21, 66 275 Rižana
Čerenka Zvonko, Mazorjev trg 1a, 66000 Koper
Bojan Štok, Gradišče 66, 62235 Gradišče
Rafko Urankar, Zadružniška 33, 61234 Mengeš

Nagradna uganka:

ŽIVI ZID

Ko sva z računalnikom prejšnjo sredo gledala nogometno tekmo, me je že na samem začetku opozoril na zanimivost. Sodnik je dosodil prosti strel in nekaj igralcev se je postavilo v živi zid. Z leve so stali takole: številka 10, številka 2, številka 5, številka 6 in nazadnje številka štiri. Štirica pa je bil po postavi največji in je po krajšem prerivanju stekel levo od desetice. Tako rekoč v istem trenutku je računalnik v spodnji levi kot zaslona nekaj izpisal. To naredi sicer le, kadar pokažejo kakšno posebno privlačno semaforko. Tokrat pa so ga ta števila zadela na njegovo računsko nožico. Ugotovil je, da je število, ki so ga sestavljale številke na majicah igralcev, postalo po premiku igralca štiri natanko štirikrat večje. Brihten računalnik, ni kaj.

Ker ga nisem takoj razumel, mi je pomagal

4 * 10256 4 = 410256

Nasprotnikovega igralca je ta zamenjava tako zmedla, da se je žoga počasi odtrkljala mimo štirice proti nasprotnikovemu golu. Tudi z računalnikom sva ... [illegible] ... od ... gledava prav vsako nogometno tekmo, še posebej pozorna pa sva takrat, ko postavljajo žive zidove. Včasih namreč kak igralec, recimo s številko "n", steče z zadnjega na prvo mesto in število, ki ga igralci sestavljajo, postane n * večje.

Nogomet imava od takrat vedno rajši, a jesenski del se izteka, zato potrebujeva vašo pomoč. Če poznate kakšnega sodnika, mu recite, naj pogosteje dosoja proste strele. Če pa imate računalnik, ki vam še ni zrasel čez glavo, naj vam pomaga. V športni redakciji se menda spomnijo dogodka, ko je z zadnjega na prvo mesto skočil igralec št. 9.

Kdo pa so bili ostali ?

Bliža se Novo leto. Dedek Mraz bo med pravilnimi rešitvami izžrebal 10 nagrad po 500 ND, 3 angleške kasete s programi in 3 celoletne naročnine na Moj mikro .

Rešitve pošljite do 15. 1. 1985 na naslov:

Uredništvo revije Moj mikro,
p p. 150-III,
61001 Ljubljana,
s pripisom "živi zid".

Trgovina z največjo izbiro
DELO
MALI OGLASI
TELEFON 061 22 33 11

VESELA VEST ZA VSE USMERJENE, STARE IN MLADE

Nikoli ni prepozno!

Mama sit sem že streljanja na marsovce računalniške igre me dolgočasijo Hočem nekaj več nekaj ustvarjalnega Hočem se naučiti programiranja da se bom lahko pogovarjal z računalnikom V novem letu se želim naučiti več kot mi daje usmerjena šola Za novoletno darilo si želim knjigo Jureta Špilerja BASIC

Z razvojem mikroelektronike so se v svetu v zadnjem času pa tudi pri nas razširili ceneni tako imenovani HIŠNI RAČUNALNIKI Te računalnike programiramo večinoma v programskem jeziku BASIC Pomanjkanje literature s tega področja zapolnjuje pričujoča knjiga, ki je namenjena najširšemu krogu bralcev Ker je danes pri nas veliko različnih računalnikov tuje proizvodnje z zelo različnimi možnostmi in se ne da vseh opisati, je v knjigi opis tistih možnosti ukazov in funkcij ki so skupne večini pri nas priljubljenih računalnikov Tako je knjiga primerna za lastnike računalnikov Apple BBC Commodore Sinclair Spectrum, Sharp Oric in drugih.

Knjiga je namenjena vsem, ki se tako ali drugače zanimajo za računalništvo Je predvsem učbenik ki začetniku omogoča da se nauči ravnati z računalnikom deloma pa tudi navodilo za uporabo in zbirka primerov

Predznanja ni treba veliko Bralec ki je končal vsaj štiri razrede osnovne šole ki zna uporabljati preprost žepni kalkulator in je samostojno prebral vsaj Martina Krpana ne bo imel dosti težav — knjiga je torej namenjena najširšemu krogu v starosti 10—100 let Za uspešno učenje moramo le kdaj pa kdaj imeti na voljo kakršenkoli hišni računalnik.

Snov je urejena po poglavjih Prva tri so uvod v računalništvo in programiranje sledi podroben opis ukazov in funkcij s primeri na koncu pa je pregled vseh ukazov tabela ASCII znakov zbirka primerov za vaje pregled hišnih računalnikov in seznam priporočene literature.

Uporabljani so predvsem že udomačeni računalniški izrazi ne pa nasilni prevodi Tako je na primer uporabljena beseda byte ki je med računalnikarji tako udomačena kot v splošni rabi na primer beseda taxi Pogosto so v oklepaju navedeni angleški izrazi za kasnejše lažje razumevanje izvirnih besedil Na koncu je še slovarček veliko uporabljanih besed.

MAMA, ZNAM BASIC RAČUNALNIK ME RAZUME!!!

Knjigo lahko kupite v knjigarni ali pa naročite po pošti na naslov: Jure Špiler, BASIC, p. p. 302, 61001 LJUBLJANA.